

UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO

CARRERA:
INGENIERÍA ELECTRÓNICA

Trabajo de titulación previo a la obtención del título de:
INGENIERO ELECTRÓNICO

TEMA:
DESARROLLO DE UNA RED IOT CON TECNOLOGÍA LORA PARA
DETECCIÓN DE AUTOMÓVILES

AUTOR:
JEAN CARLOS BUESTÁN VERA

TUTOR:
JOSÉ RENATO CUMBAL SIMBA

Quito, febrero 2019

CESIÓN DE DERECHOS DE AUTOR

Yo Jean Carlos Buestán Vera , con documento de identificación N° 171569343-6, manifiesto mi voluntad y cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del trabajo de titulación intitulado: "DESARROLLO DE UNA RED IOT CON TECNOLOGÍA LORA PARA DETECCIÓN DE AUTOMÓVILES.", mismo que ha sido desarrollado para optar por el título de Ingeniero Electrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.



Jean Carlos Buestán Vera
C.I.: 171569343-6

Quito, febrero 2019.

DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR

Yo declaro que bajo mi dirección y asesoría fue desarrollado el trabajo de titulación, **DESARROLLO DE UNA RED IOT CON TECNOLOGÍA LORA PARA DETECCIÓN DE AUTOMÓVILES**, realizado por Jean Carlos Buestán Vera, obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana para ser considerado como trabajo final de titulación.

Quito, febrero del 2019.



José Renato Cumbal Simba

CI: 1714663760

DEDICATORIA

En primer lugar, quiero agradecer a los pilares de este logro importante que son mis padres Víctor Eduardo Buestán Toral y Amelia de Jesús Vera Moreira, porque al brindarme el apoyo incondicional en cada paso de mi vida y sobretodo brindarles la oportunidad de que vivan conmigo, un logro más como es obtener una ingeniería; además a toda mi familia que fueron el aliento principal al seguir adelante con más énfasis para terminar este trabajo de titulación.

En segundo lugar, a la Virgencita “DOLOROSA”, por guiarme por el camino de la fe y de la sabiduría en cada instante para encontrar la solución a cada adversidad tanto académica como espiritual. Al Señor Todopoderoso por siempre escuchar mis oraciones para seguir adelante en medio de tantos tropiezos en mi vida y saberlos afrontar con madurez y dedicación.

AGRADECIMIENTO

Agradezco a la Universidad Politécnica Salesiana por formarme con valores y brindarme los conocimientos necesarios para afrontar la vida de forma profesional, a los docentes que se tomaron el tiempo de enseñarnos habilidades fuera de las materias e impartir las soluciones ingeniosas a cada dificultad.

Un agradecimiento especial al ingeniero José Renato Cumbal Simba y al ingeniero Juan Carlos Domínguez Ayala por colaborarme en cada instante del trabajo para culminar el proyecto tanto escrito como hardware y tener la facilidad de acceso a redes, específicamente, orientadas hacia la red IoT.

A su vez agradecer al Departamento de Soporte Técnico y Tecnología de la Universidad Politécnica Salesiana por brindar las soluciones de conexión para el proyecto elaborado.

ÍNDICE GENERAL

CESIÓN DE DERECHOS DE AUTOR.....	i
DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR.....	ii
DEDICATORIA	iii
AGRADECIMIENTO	iv
ÍNDICE GENERAL.....	v
ÍNDICE DE FIGURAS.....	vii
ÍNDICE DE TABLAS	x
RESUMEN.....	xi
ABSTRACT.....	xii
INTRODUCCIÓN	xiii
CAPÍTULO 1	1
ANTECEDENTES.....	1
1.1 Planteamiento del Problema.....	1
1.2 Justificación.....	2
1.3 Objetivo.....	3
1.3.1 Objetivo General.....	3
1.3.2 Objetivo Especifico.....	3
CAPÍTULO 2	4
FUNDAMENTACIÓN TEÓRICA.....	4
2.1 Internet de las cosa (IoT).....	4
2.2 LORAWAN.....	5
2.3 Topología LORAWAN.....	6
2.4 Clase LORAWAN.....	6
2.5 RED DE SENSORES INALAMBRICOS.....	7
2.5.1 Modulación LoRa.....	8
2.5.2 Protocolo LoRaWAN.....	8
2.5.3 LORA/GPS SHIELD.....	9
2.6 Seguridad LoRaWAN.....	10
2.7 CONECTIVIDAD IOT AUTOMOVILES.....	10
2.8 (IoT) LPWAN	11
2.9 GPRS SIM 900.....	12

CAPÍTULO 3.....	3
DISEÑO E IMPLEMENTACIÓN.....	13
3.1 Requerimientos para implementar la red IoT (Internet de las cosas).....	15
3.2 Mediciones Previas en el vehículo.....	15
3.3 Explicación del tramo elegido y Aplicación de Modelo Matemático Okumura-Hata para el área suburbana.....	24
3.4 Formato de la Trama Física de LoRa.....	27
3.5 Registro de datos mediante una doble cámara incorporada en el vehículo.....	33
CAPÍTULO 4.....	34
PRUEBAS Y RESULTADOS.....	34
4.1 Manejo de la plataforma ThingSpeak.....	34
4.2 Funcionamiento de la Pagina WEB.....	37
4.3 Representación de los datos adquiridos en las plataformas Usadas.....	39
CONCLUSIONES	47
RECOMENDACIONES.....	51
REFERENCIAS.....	52
ANEXOS	57

ÍNDICE DE FIGURAS

Figura 2.1 Arquitectura del Internet de las Cosas (IoT).....	4
Figura 2.2 Arquitectura de red LoRaWAN.....	6
Figura 2.3 Dispositivo Final Clase A.....	7
Figura 2.4 Componentes de WNS Gateway y nodos distribuidos	8
Figura 2.5 Protocolo LORAWAN	8
Figura 2.6. Lora/GPS.....	9
Figura 2.7. Flujo de datos LORAWAN.....	10
Figura 2.8 Aplicaciones de la IOT con tecnología de conectividad.....	11
Figura 2.9 Comparación y Estadística de las redes de comunicación Local o Inalámbrica.....	11
Figura 2.10 Representación LPWAN, ZIGBEE, 3G/4G/5G en la red IOT.	12
Figura 2.11 GPRS SIM900.....	12
Figura 3.1 Diagrama de flujo para manejar distintas variables a controlar con dispositivos LoRa.....	13
Figura 3.2. Patrón Lineal de la investigación cuantitativa.....	14
Figura 3.3 Adquisición de valores en voltaje de los sensores de temperatura y Velocidad.....	16
Figura 3.4 Extracción del Reóstato para medir el nivel de Gasolina.....	17
Figura 3.5 Reóstato extraído del depósito de la gasolina.	18
Figura 3.6. Valor tomado de Reóstato en ohmios.....	19
Figura 3.7 Velocidad evaluada con Scanner automotriz.....	20
Figura 3.8 Relación Kilómetros Vs Voltaje.....	20
Figura 3.9 Medida del valor de la resistencia del sensor de temperatura ECT tipo NTC.....	21
Figura 3.10 Módulo Shield LoRa GPS y Módulo Repetidor Shield de LoRa – GPRS.....	21
Figura 3.11 Sensores en el vehículo.....	22
Figura 3.12 Señal adquirida desde el tanque del vehículo.....	23
Figura 3.13 Módulo Shield GPS LoRA en el vehículo.....	23
Figura 3.14 Repetidor con tecnología GSM- GPRS SIM 900.....	24

Figura 3.15 Trayectoria Optada para realizar la implementación de la Red.....	24
Figura 3.16 Trayectoria simulada según la arquitectura propuesta por LoRaWAN.....	25
Figura 3.17. Factor de corrección ciudades grandes.....	26
Figura 3.18. Formato de la trama física.....	27
Figura 3.19. Dragino Shield con pines para conexión con Arduino.....	28
Figura 3.20. Dragino LoRA/GPS Shield con pines selectivos para Arduino.....	29
Figura 3.21. Arquitectura Gateway LG01.....	30
Figura 3.22. Interfaz de acceso al conectar con la dirección IP 10.130.1.1.....	31
Figura 3.23. Selección de puerto Dragino-169d30 sobre la IP: 10.130.1.1 en Arduino Yun.....	31
Figura 3.24. Dispositivos LoRa para realizar comunicación entre cliente y servidor.....	32
Figura 3.25. Comunicación LoRA en el Gateway LG01.....	32
Figura 3.26 Verificación mediante imágenes del trayecto elegido con los parámetros de velocidad, GPS satelital y coordenadas.....	33
Figura 4.1 Interfaz de la plataforma ThingSpeak.. ..	34
Figura 4.2. Arquitectura para una red IoT con la plataforma ThingSpeak.....	35
Figura 4.3. Configuración del canal de ThingSpeak con 6 campos de lectura.....	36
Figura 4.4. Configuración entre ThingSpeak y ARDUINO mediante el canal de identificación.....	36
Figura 4.5. Interface de la Pagina WEB.....	38
Figura 4.6. Parametros adquiridos en ThingSpeak.....	39
Figura 4.7. Representación de los parámetros adquiridos en base a la temperatura, latitud y longitud.....	40
Figura 4.8. Acceso WEB para iniciar sesión con el usuario.....	40
Figura 4.9. Representación de los parámetros registrados por el servidor de la WEB.....	41
Figura 4.10. Representación individual de los parámetros medidos por la WEB.....	42
Figura 4.11. Ubicación exacta de la posición actual del vehículo.....	43
Figura 4.12 Recepción de datos por los sensores del vehículo mediante el monitor serial.....	44

Figura 4.13 Monitor serial del envío de datos a través del GPRS a la WEB.....	45
---	----

ÍNDICE DE TABLAS

Tabla 1.1. Desarrollo comparativo de tecnología de largo alcance para la IoT.....	5
Tabla 3.1. Valores tomados de la gasolina en Voltaje.....	18
Tabla 3.2 Análisis de la Velocidad y Amplitud tomada con un Scanner Automotriz.....	19
Tabla 3.3 Tipos de mensajes MAC.....	28
Tabla4.1. Reporte de los Sensores LoRa montados en el vehículo.....	43
Tabla4.2. Costo de Implementación red IoT.....	46

RESUMEN

El presente proyecto busca una nueva alternativa para conectar las redes inalámbrica con el internet de las cosas (IoT), optimizando tanto el levantamiento de la información, transmisión y recepción de datos en tiempo real, facilitando la visualización de los datos adquiridos mediante una base de datos y generando reportes históricos que aportan en la supervisión de un vehículo al transitar por un tramo de difícil acceso en el ámbito de conectividad.

Existen en el mercado varias tecnologías que van de la mano con el internet de las cosas y de largo alcance, gran cantidad de conexiones inalámbricas ya probadas para implementarse dentro de un área específica, por tal razón se quiere demostrar que la tecnología LoRa/LoRAWAN tiene una ventaja dentro de las nuevas tecnologías inalámbricas.

A través de un Gateway se pretende recopilar los datos adquiridos mediante los sensores básicos de un automotor como es la temperatura, velocidad, nivel de gasolina, nivel de agua y monitoreo GPS. Mediante los Shield de LoRa se quiere brindar la posibilidad de recopilar los parámetros en una base de datos propia de la IoT, a su vez se quiere proporcionar al usuario no solo una plataforma de supervisión sino una alternativa como es el manejo propio de una página WEB para tener un reporte más detallado de los valores adquiridos mediante los equipos implementados en el vehículo.

Al trabajar con datos que se actualiza cada 15 segundos se tiende a perder información valiosa por tal razón se quiere afianzar la conectividad usando la red GPRS.

ABSTRACT

This project seeks a new alternative to connect wireless networks with the Internet of Things (IoT), optimizing both the gathering of information, transmission and reception of data in real time, facilitating the visualization of data acquired through a database and generating historical reports that provide in the supervision of a vehicle to pass through a stretch of difficult access in the field of connectivity.

There are several technologies in the market that go hand in hand with the internet of things and the long reach, large amount of wireless connections and solutions to be implemented within a specific area, therefore, the LoRa / LoRAWAN technology has an Advantage within the new wireless technologies.

Through a Gateway it is intended to collect the data acquired through the basic sensors of a car such as temperature, speed, fuel level, water level and GPS monitoring. Through the LoRa Shield, we want to offer the possibility of collecting the parameters in a database of the IoT, and this time we want to provide a user not only a monitoring platform but an alternative such as the proper management of a WEB page for have a more detailed report of the values acquired through the equipment implemented in the vehicle.

When working with data that is updated every 15 seconds, it is seen, lost, valued, justified, strengthened, the connectivity is used the GPRS .

INTRODUCCIÓN

La constante expansión de las redes inalámbricas y el crecimiento que se viene dando en el desarrollo del campo de las comunicaciones móviles ha motivado la búsqueda de nuevas soluciones de conectividad. Dicho crecimiento ha sido continuo y se ha dado a paso acelerado, en tal medida es que se ha desarrollado la tecnología IoT que se centra en LoRaWAN (acrónimo de Long Range Wide-area network) y en LPWAN (en inglés, *Low Power Wide Area Network*), redes de largo alcance y bajo consumo.

Paulatinamente la humanidad se ha dado cuenta que todo equipo o conexión en sí, se está orientando para a ser inalámbrica, lo cual permite minimizar el uso de cables para obtener información, como a su vez tener conectado todo el momento equipos para que funcionen sin interrumpir su monitoreo. El desarrollo de la nueva tecnología de largo alcance e interconectividad inalámbrica soluciona varios inconvenientes en el enlace entre equipos y sobretodo en la evolución continua con nuevos dispositivos.

La tecnología LoRa abarca la disponibilidad de trabajar en distintas frecuencias y áreas en las cuales no existe una cobertura fija, la tecnología también engloba el estudio y su implementación en varias zonas (urbana, rural y suburbana). Por ejemplo, en su implementación, esta tecnología puede trabajar en el área meteorológica, área de cultivos, rastreos, estudios del ambiente, área de la medicina y también en el área ecológica. El aporte significativo de esta tecnología es trabajar con redes de la IoT y equipos de bajo consumo (hasta 10 años con una batería), con ello los sensores no necesitan corriente eléctrica de red, y al ser una comunicación de radio frecuencia no necesita de licencias ya que utiliza el espectro de la banda de radio ISM, que tiene aplicaciones en el ámbito industrial, científico y médico.

Posteriormente, se realizará el estudio pertinente y la aplicación de LoRA/LoRaWAN con los equipos necesarios para poder limitar o favorecer su uso dentro de la Autopista SIMÓN BOLÍVAR, en donde se implementará una red IoT que permitirá monitorear, sin pérdida de datos, las variables adquiridas dentro de un vehículo en continuo movimiento sin interrumpir la conexión.

Es por esto que la tecnología implementada nos brinda la posibilidad de tener dicha adquisición a largo alcance. Y específicamente, en esta investigación se van a analizar

los siguientes factores en los vehículos: su localización, la temperatura, lectura en medición de combustible, velocidad del vehículo, nivel de combustible, nivel de agua y a su vez una vista satelital del trayecto optado para las mediciones mencionadas.

CAPÍTULO 1

ANTECEDENTES

1.1 Planteamiento del Problema

El mundo de las redes inalámbricas ha evolucionado de tal forma que existen diferentes tipos de conexiones de largo alcance. Los diseñadores en la actualidad buscan la manera óptima de conectar tecnologías inalámbricas con un producto a internet. En tal caso, bajo la presente investigación se quiere resolver con una aplicación la problemática en donde existen tramos de poca cobertura y a su vez pocas conexiones inalámbricas.

Dentro de las autopistas, existen varios inconvenientes y factores al momento de conducir un vehículo sin tener idea el estado del transporte utilizado. En la actualidad hay varias formas de mantener la estabilidad vehicular, en el caso de supervisar los parámetros mecánicos básicos y que el usuario esté al tanto de cómo se encuentra el automotor a utilizar. Por ejemplo existen maneras de monitorear, pero que no se tiene al tanto del entorno detallado de ubicación, niveles de líquidos, temperatura y velocidad real a la cual transita, por el momento la tecnología que tiene más relevancia es la inalámbrica, esta supervisión se la realizaba y se realiza en base a enlaces satelital y pocas veces mediante cobertura celular; por ende la idea es brindar e implementar una nueva tecnología inalámbrica que abarca la posibilidad de solucionar los problemas de cobertura y supervisión general de la aplicación a trabajar. Al proponer una nueva tecnología en base a radio frecuencia se habla de la posibilidad de implementar nuevas soluciones para la IoT, se pretende desarrollar una red inalámbrica que cuente con tecnología LoRa y mediante el uso de sensores poder monitorear el estado del vehículo al momento de transitar por un tramo elegido. Puede que la autopista no proporciona una buena conectividad celular, pero en este caso se propone solucionar dicho inconveniente mediante una arquitectura y herramientas desarrolladas por LoRa y la IoT.

El prototipo que se va a desarrollar pretende brindar una clara opción en usar nuevas tecnologías como es LoRaWAN. Este permitirá al usuario tener un control exacto del estado del automotor que transita día a día y así evitar tener complicaciones como en

el caso de encontrarse en una autopista que no tiene una amplia cobertura para solicitar un servicio si existen complicaciones. La red desarrollada por la IoT abarca la posibilidad de implementar el prototipo de red con éxito y a su vez verificar que es una conexión ideal para realizar en varios nodos ya que en este caso se implementara en un nodo con varios sensores implementados.

1.2 Justificación

La innovación que se va a introducir en la presente monografía es LoRaWAN, ya que apunta a requerimientos característicos del Internet de las Cosas, tales como “conexiones bidireccionales seguras, bajo consumo de energía, largo alcance de comunicación, bajas velocidades de datos, baja frecuencia de transmisión, movilidad y servicios de localización” (Efor, s.f.). En tal sentido, esta nueva tecnología LoRa brinda a operadores de redes el manejo de aplicaciones de baja potencia, red de área amplia y soluciones de bajo costo que se pueden adaptar con dispositivos de alta autonomía. Todo esto busca ser una innovación que brinde mayor comodidad al usuario para tener un monitoreo seguro y continuo, ya que a veces el automotor no es supervisado de la mejor manera en el ámbito de la mecánica básica. El éxito de implementar una alternativa de conexión a largo alcance es dar a conocer a la sociedad que se trabaja y se puede proponer nuevas opciones para desarrollar el confort al momento de utilizar un vehículo.

Al hablar de desarrollar una red con el internet de las cosas se pretende dar a conocer las herramientas de suma importancia que se ha dado en ciudades inteligentes, ya que equipos de radio frecuencia es solo utilizado en ciertos ámbitos tecnológicos, pero el aporte como tal permite dar a conocer que una red diseñada e implementada es capaz de comunicar e interactuar con sensores a una larga distancia de supervisión.

La interoperabilidad entre dispositivos inteligentes es una de las ventajas claras para solucionar e implementar varios repetidores y tratar de comunicar punto a punto varios sensores inalámbricos que me permitirán entrar en varios ámbitos.

La importancia de desarrollar este tipo de red es demostrar en base al prototipo como tal que se puede implementar en varios nodos, en este caso el prototipo es supervisar en un solo nodo varios sensores para los parámetros básicos de un automóvil en base a la tecnología de espectro ensanchado ya que me permitirá manejar velocidades de

datos que no interfieren con otras comunicaciones, la relevancia de este estándar es la red LoRaWAN porque, no necesita instalaciones locales ya está abierta a muchas aplicaciones importantes.

1.3 Objetivos

1.3.1 Objetivo General

Desarrollar una red IOT con tecnología LoRa para la ubicación de automóviles y medición de algunos parámetros como cantidad de combustible, temperatura, agua.

1.3.2 Objetivos Específicos

- Analizar los distintos tipos de redes IOT de largo alcance para identificación de parámetros de vehículos en carreteras.
- Determinar un sistema de adquisición de datos en vehículos para la obtención de parámetros en carreteras.
- Implementar una red prototipo IOT para la determinación de los principales parámetros de un vehículo y verificar mediante imágenes.
- Analizar la máxima distancia en que se puede implementar la red IOT para determinar hasta que distancia se reciben los datos correctamente.
- Analizar los costos para la implementación de la red IOT.

CAPÍTULO 2

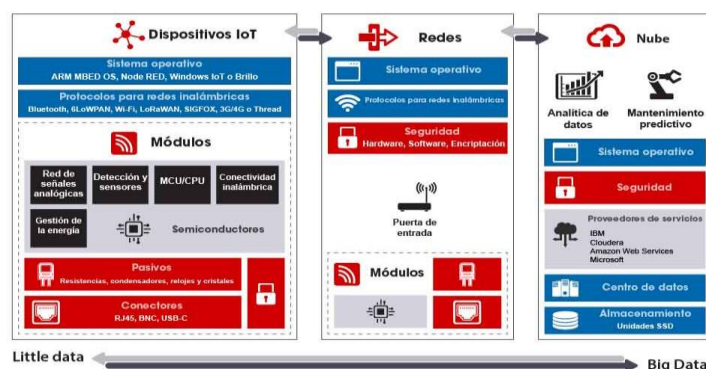
FUNDAMENTACIÓN TEÓRICA

2.1 Internet de las cosas (IoT)

Desde hace algunos años que se comenzó a hablar sobre el concepto de “Internet of things” (Iot) o en su traducción al español, el “internet de las cosas”. Si bien en sus comienzos era una propuesta a futuro, hoy en día, mediante el desarrollo de nuevas tecnologías vemos como este se está convirtiendo en una realidad. Si bien existen varias definiciones sobre el Iot, consideramos pertinente entenderlo como: “Un sistema de dispositivos de computación interrelacionados, máquinas mecánicas y digitales, objetos, animales o personas que tienen identificadores únicos y la capacidad de transferir datos a través de una red, sin requerir de interacciones humano a humano o humano a computadora”. (SAP, 2016)

En base a la definición anterior se debe entender que para el Internet de las cosas, una cosa, puede ser: “El estado de los semáforos mediante el uso del internet, si se encuentran funcionando de manera óptima al momento de dar las señales de tránsito. También mediante sensores, tenemos la oportunidad de saber parámetros básicos de un vehículo al momento de transitar como es el estado de los neumáticos, nivel de gasolina, velocidad (rpm) y la cantidad de luz en el día para ir aumentando la luminosidad de los faros cada vez que se oscurezca la tarde”

Figura 2.1 Arquitectura del Internet de las Cosas (IoT).



Arquitectura de la IoT orientada desde el manejo de pequeños datos a grandes datos en base a módulos, dispositivos, redes y manejo de datos en la nube. Fuente:rs-online.

A continuación, tenemos los diferentes tipos de redes IoT orientados a un largo alcance y óptimos para implementar en la supervisión del vehículo. Mediante una tabla

comparativa se tendrá en cuenta la relevancia de la tecnología LoRa ante importantes tecnologías inalámbricas:

Tabla 1.1 Desarrollo comparativo de tecnología de largo alcance para la IoT.

Protocolos inalámbricos para IOT	Frecuencia	Gama (entorno no urbano)	Velocidad de datos	Topología	Requiere Hub o Gateway?	¿Propietario o Abierto?
ZIGBEE	2.4GHz, 915MHz (EE.UU.), 868 MHz (UE)	100-325 pies	250 kbps (2.4) 40kbps (915) 20kbps (868)	Malla	Sí	Abierto
WIFI	2.4GHz / 5GHz	115-230	7Gbps	Estrella	No	Abierto
Wi-Fi-ah (HALOW)	900MHz	3000 pies	347Mbps	Estrella	No	Abierto
Hilo	2.4GHz	100 pies	250kbps	Malla	Sí	Abierto
DIGIMESH	2.4GHz / 900 MHz (EE. UU.) / 868 MHz (UE)	~ 20 millas	250 kbps (2.4) 40kbps (915) 20kbps (868)	Malla	Sí	Propiedad
MIWI	2.4GHz o SUBGHZ	800 pies	250kbps	Malla o estrella	Sí	Propiedad
ENOCEAN	900Mhz (US) 868 MHz (EU) 315 MHz	30-100 pies	125 kbps	Malla	Sí	Propiedad
6LoWPAN	2.4GHz	380 pies	250kbps	Malla	Sí	Abierto
Sin peso (W, N, P)	espacios en blanco, 915 MHz, 868 MHz, 780 MHz, 470 MHz, 433 MHz, 169 MHz	1.2 millas (P), 3 millas (W, N)	200bps-100kbps	Estrella	Sí	Abierto
MCTHING	2.4GHz	650 pies	50kbps	Estrella	Sí	Propiedad
LORA	150MHz-1GHz (muchas opciones)	hasta 20 millas	50kbps	Estrella	Más o menos	Abierto
SIGFOX	900Mhz (US) 868 MHz (EU)	~ 20 millas	100bps	Estrella	Sí	Propiedad
LTE Cat-M1	1.4MHz	~ 20 millas	1 Mbps	Estrella	No	Abierto
CAT-M2	Por debajo de 1 GHz	~ 20 millas	100kbps	Estrella	No	Abierto
3G y 4G celular (US)	700 MHz, 800 MHz, 850MHz, 1700MHz, 1900MHz, 2100MHz, 2300MHz, 2500MHz	~ 20 millas	200kbps (3G) 10Mbps (4G)	Estrella	No	Abierto

Desarrollo comparativo de tecnología de largo alcance para la IoT. Elaborado por: Jean Carlos Buestán.

Dentro del desarrollo vehicular, en base a una tecnología inalámbrica con varios factores favorables, involucra brindar al usuario la accesibilidad y fiabilidad al momento de implementar cualquier tecnología, teniendo una correcta caracterización de cada conexión inalámbrica podemos observar en la Tabla 1.1

2.2 LoRaWAN

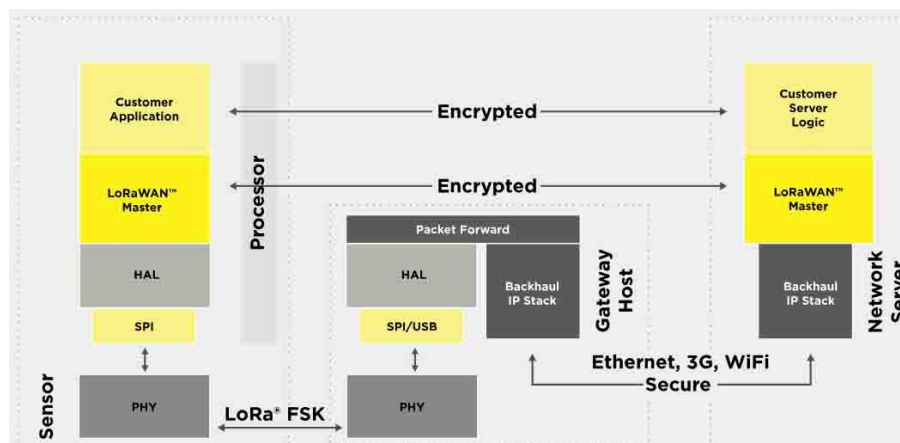
La especificación LoRaWAN™ es un protocolo de red de área amplia de baja potencia (LPWA) diseñado para conectar de forma inalámbrica "cosas" operadas por batería a Internet en redes regionales, nacionales o globales. El protocolo incluye funciones que admiten comunicaciones bidireccionales de bajo costo, móviles y seguras para Internet de las cosas (IoT), máquina a máquina (M2M), ciudad inteligente y aplicaciones

industriales. (Alliance, 2017)

2.3 Topología LoRaWAN

La especificación define los parámetros de la capa física del dispositivo a la infraestructura (LoRa®) y el protocolo (LoRaWAN) y, por lo tanto, proporciona una interoperabilidad perfecta entre los fabricantes, como se demostró a través del programa de certificación del dispositivo. Las puertas de enlace están conectadas al servidor de red a través de conexiones IP estándar y actúan como un puente transparente, simplemente convirtiendo paquetes RF en paquetes IP y viceversa. (ALLIANCE L. , 2017)

Figura 2.2 Arquitectura de red LoRaWAN



Topología en la que las puertas de enlace retransmiten mensajes entre los dispositivos finales y un servidor de red central. Fuente: lora-alliance.

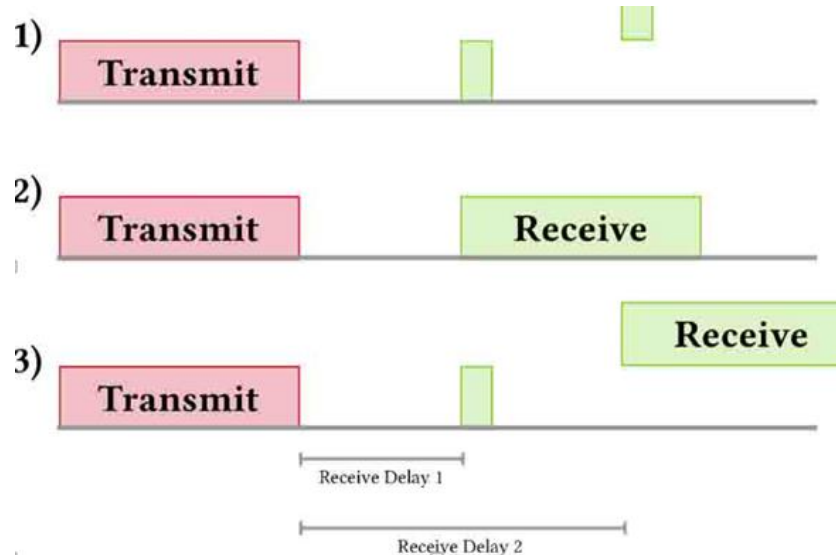
2.4 Clases LoRaWAN

LoRaWAN tiene varias clases diferentes de dispositivos end-points para atender las diferentes necesidades en la amplia gama de aplicaciones, estos se especificarán a continuación:

- **Dispositivos finales bidireccionales (clase A):** Permiten tener una comunicación a través de una transmisión mediante dos ventanas de recepción de manera ascendente y por lo cual lo más esencial que son bidireccionales con una pequeña variación basada en una base de tiempo aleatoria. (tipo de protocolo ALOHA) (MULTITECH, 2018)

- **Dispositivos finales bidireccionales con ranuras de recepción programadas (Clase B):** La gran ventaja que proporciona la clase B es la recepción a través de ventanas a horas programadas y a su vez recibir una notificación para saber cuándo el dispositivo final está escuchando. (MULTITECH, 2018)
- **Dispositivos finales bidireccionales con ranuras de recepción máximas (Clase C):** En la recepción de datos tiene la capacidad de estar continuamente abierta y cerrarse cuando está transmitiendo. (MULTITECH, 2018)

Figura 2.3 Dispositivo Final Clase A



El servidor puede responder en la primera ventana de recepción o en la segunda ventana de recepción, pero no debe usar ambas ventanas. Fuente: thethingsnetwork

2.5 Red de sensores inalámbricos

Se define como una red auto-configurable “integrada por un pequeño número de nodos sensores también llamados motes distribuidos espacialmente y comunicados entre sí para lo que se usan señales de radio, con la finalidad de monitorizar y entender el mundo físico”. (Archila & Santamaria, 2013)

Figura 2.4 Componentes de WNS Gateway y nodos distribuidos.



Red de Sensores inalámbricos. Fuente: Instrument,N.

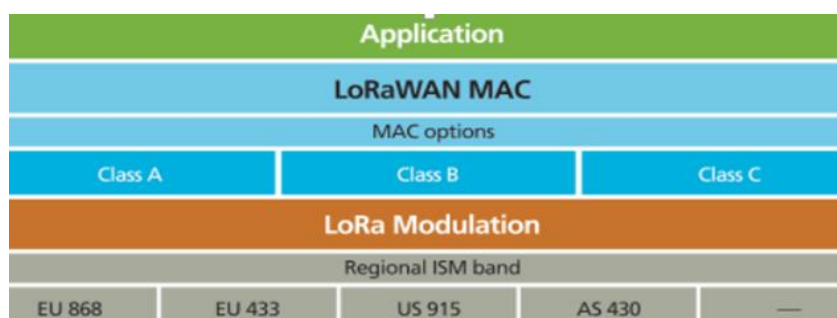
2.5.1 Modulación LoRa

La modulación FSK es utilizada en una gran variedad de sistemas inalámbricos principalmente porque es muy eficaz para lograr baja potencia. No obstante, “LoRa está basada en la Modulación CSS (Chirp Spread Spectrum) que ha sido utilizada para comunicaciones espaciales y militares por décadas debido a larga distancias de comunicación que se puede lograr y robustez a la interferencia. LoRa es la primera en implementar esta modulación de forma comercial”. (Burbano, 2017, pág. 18)

2.5.2 Protocolo LoRaWAN

“LoRaWAN es un protocolo de red que usa la tecnología LoRa para comunicar y administrar dispositivos LoRa” (Medium, 2018), este a su vez, consta de dos partes principales, las cuales son: “gateways y nodos, los primeros son los encargados de recibir y enviar información a los nodos y los segundos, son los dispositivos finales que envían y reciben información hacia el gateway”. (Medium, 2018)

Figura2.5. Protocolo LORAWAN



Aplicaciones en base al protocolo, modulación, regional y opciones MAC. Fuente: SEMTECH.

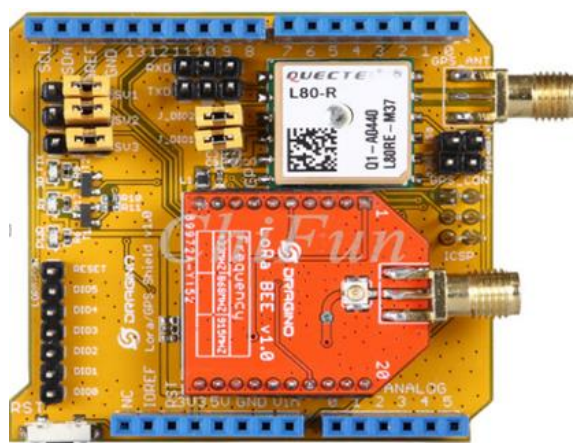
En resumen, dentro de todas las características de LoRaWAN, las principales y las más destacadas para nuestros fines investigativos, y tal como lo plantean desde el portal Medium (Medium, 2018), son las siguientes:

- Topología estrella
- Alcance de 10 a 15km en línea de vista
- Soporte para 3 clases de nodos
- Administración de dispositivos
- Redes públicas y privadas
- Bajo consumo y largo alcance
- Baja transferencia de datos (hasta 242 bytes)

2.5.3 LORA/GPS SHIELD

Lora/GPS se basa en el SX1276/SX1278 transceptor. Los transceptores de Lora/Shield GPS, característica la Lora MODEM de largo alcance, proporcionan un ultra-largo alcance de comunicación de espectro ensanchado, y alta resistencia a las interferencias, con lo cual se va reduciendo al mínimo el actual consumo. Proporciona también importantes ventajas tanto en el bloqueo y técnicas de modulación de selectividad más convencionales, esto lleva a resolver el tradicional diseño de compromiso entre el rango, inmunidad a las interferencias y el consumo de energía.

Figura 2.6 Lora/GPS

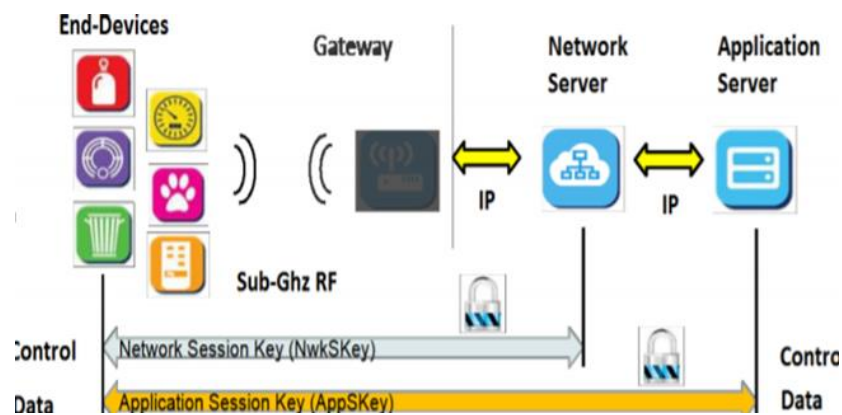


SX1276/SX1278 transceptor GPS, proporciona ultra- largo alcance. Fuente: arduinoshopping.

2.6 Seguridad LoRaWAN

En cuanto al componente de seguridad de LoRaWAN, encontramos que esta utiliza “2 capas de seguridad: uno para la red y otro para la aplicación. Network Session Key: se asegura de que la autenticidad del nodo. Application Session Key: Se asegura de que el operador tiene permiso para manejar la red” (Burbano, 2017, pág. 12). A su vez, también “se emplea la encriptación AES-128 junto con el intercambio de claves utilizando un Identificador IEEE EU164” (Burbano, 2017).

Figura 2.7 Flujo de datos LORAWAN

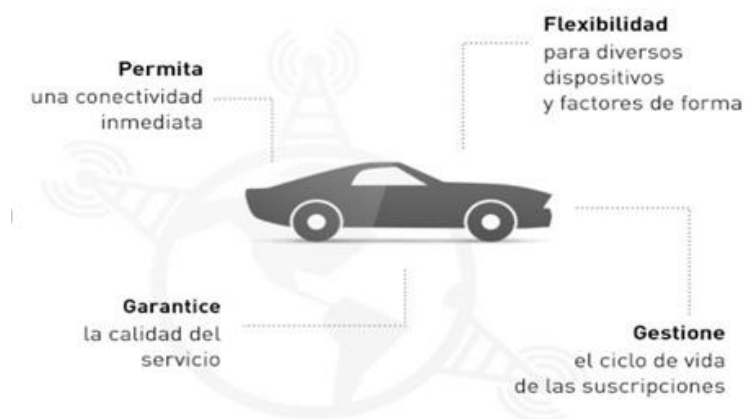


Redes y aplicaciones en el flujo, control de datos de LoRaWAN. Fuente: lora-alliance.

2.7. Conectividad IOT automóviles

La búsqueda para que la conectividad tenga una mayor presencia y sea ininterrumpida, ha impulsado el desarrollo de las tecnologías móviles, y con ello también, se ha hecho realidad el camino para la innovación en relación al Internet de las Cosas. A su vez, gracias a estos desarrollos, el surgimiento de nuevas tecnologías, así como el desarrollo e optimización en tecnologías preexistentes, en el campo de conectividad es latente. Entre estas podemos destacar a modo de ejemplo: Lora, ZigBee o BTLE, tecnologías que atenderán las necesidades orientadas a solucionar inconvenientes a nivel de conexiones. Para tales efectos, en la aplicación que se busca desarrollar en el presente proyecto, se basa en el desarrollo en un vehículo mediante la conexión inalámbrica de LORA. Este tipo de tecnología como se puede observar en la figura 2.8 presenta una serie de aplicaciones positivas en cuanto a la conectividad de automóviles.

Figura 2.8. Aplicaciones de la IOT con tecnología de conectividad.



Con LoRa la conectividad es confiable para determinar varios parámetros a identificar en lo que abarca las características de un automóvil.

2.8 (IoT) LPWAN

Las redes LPWAN no buscan necesariamente brindar velocidades más rápidas para las aplicaciones más exigentes; “al contrario, el objetivo es proporcionar una capa de baja potencia para dispositivos pequeños como sensores y medidores inteligentes. La mayoría de los usos se están encontrando en el sector industrial, mientras que las tecnologías de acceso como Bluetooth, Zigbee y Wi-Fi están orientando su explotación a aplicaciones más exigentes”. (IoTenCastellano, 2018)

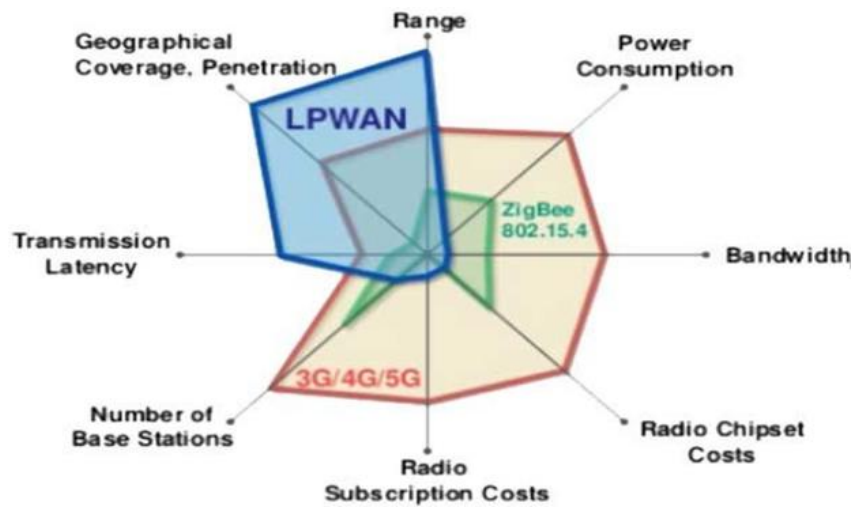
Figura 2.9. Las redes de comunicación Local o Inalámbrica.

	Local Area Network Short Range Communication	Low Power Wide Area (LPWAN) Internet of Things	Cellular Network Traditional M2M
	40%	45%	15%
😊	Well established standards In building	Low power consumption Low cost Positioning	Existing coverage High data rate
😞	Battery Live Provisioning Network cost & dependencies	High data rate Emerging standards	Autonomy Total cost of ownership
	Bluetooth 4.0, Wi-Fi	LoRa	3G, 4G

Comparación y Estadística de las redes de comunicación Local o Inalámbrica. Fuente: Open Source Iberia.

En la Figura 2.9, podemos tener una efectividad estadística de LORA del 45% de bajo consumo y costo en estándares emergentes.

Figura 2.10 Representación LPWAN, ZIGBEE, 3G/4G/5G en la red IOT



LPWAN converge orientándose a la zona geográfica entre el rango y la transmisión de latencia en la formación de la IOT. Fuente: Open Source Iberia.

En la Figura 2.4 , tenemos la posibilidad de tener un amplio conocimiento del trabajo de las tecnologías inalámbricas. LPWAN tiene la ventaja ante los demás de tener mayor rango, transmisión y convergencia geográfica para trabajar ante ZigBee y conectividad 3G/4G/5G.

2.9 GPRS SIM 900

El GPRS SIMCOM SIM900 GSM Quad band GSM shield, se describe Como: Una tarjeta ultra compacta que me permite tener una conexión celular con cualquier operadora, mediante el siguiente enlace se puede encontrar el Datasheet de la tarjeta: <https://www.robotshop.com/media/files/PDF/gprs-shield-sld33149p.pdf>

Figura 2.11. GPRS SIM900



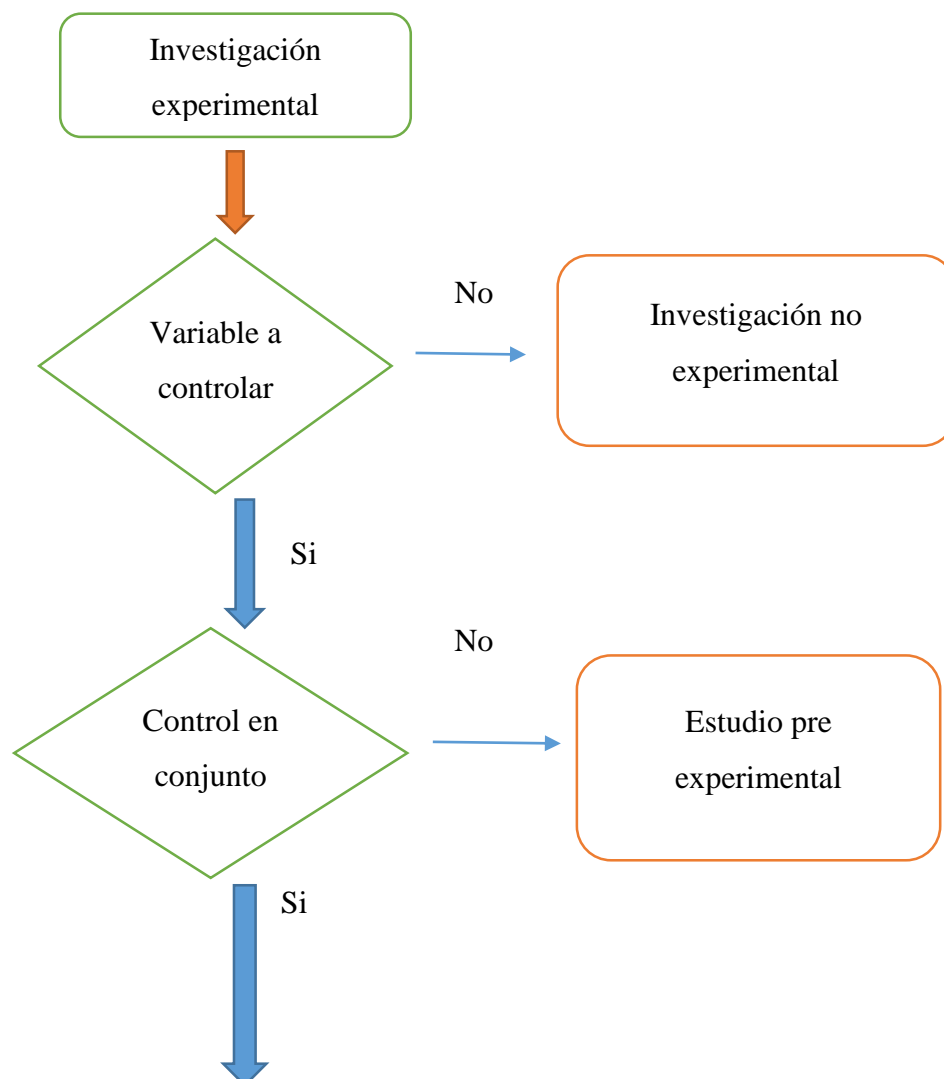
Mediante la tarjeta GPRS se procede a atenuar la conexión a internet y tener un monitoreo en tiempo real con los parámetros del vehículo. Fuente: Hetpro.

CAPÍTULO 3

DISEÑO E IMPLEMENTACIÓN

En el presente capítulo, se describe el desarrollo del hardware y software del proyecto. Se muestra el desarrollo para implementar una red IoT con la tecnología LoRa, el diseño y la arquitectura, la interconexión de los diferentes dispositivos instalados, la transmisión de datos a dos tipos de plataformas virtuales, y el monitoreo de los diferentes sensores implementados. En este caso, se manejan varios tipos de variables mediante la configuración de dispositivos LoRa, representados mediante diagrama de flujo, lo cual describirá la evidencia causa-efecto en lo que se refiere a mayor alcance y consumo de energía.

Figura 3.1 Diagrama de flujo para manejar distintas variables a controlar con dispositivos LoRa.

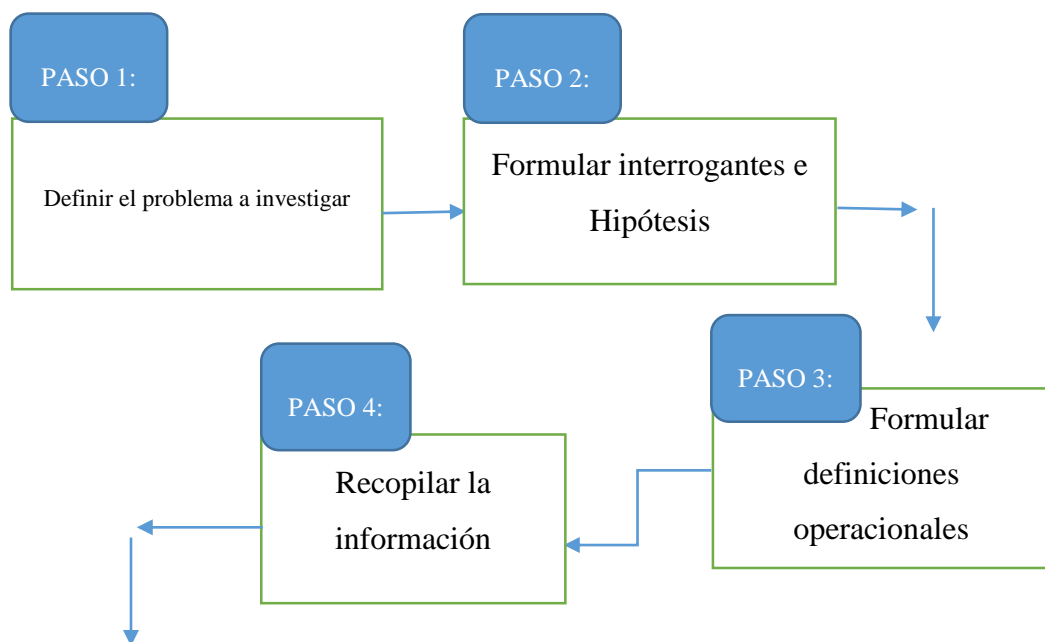


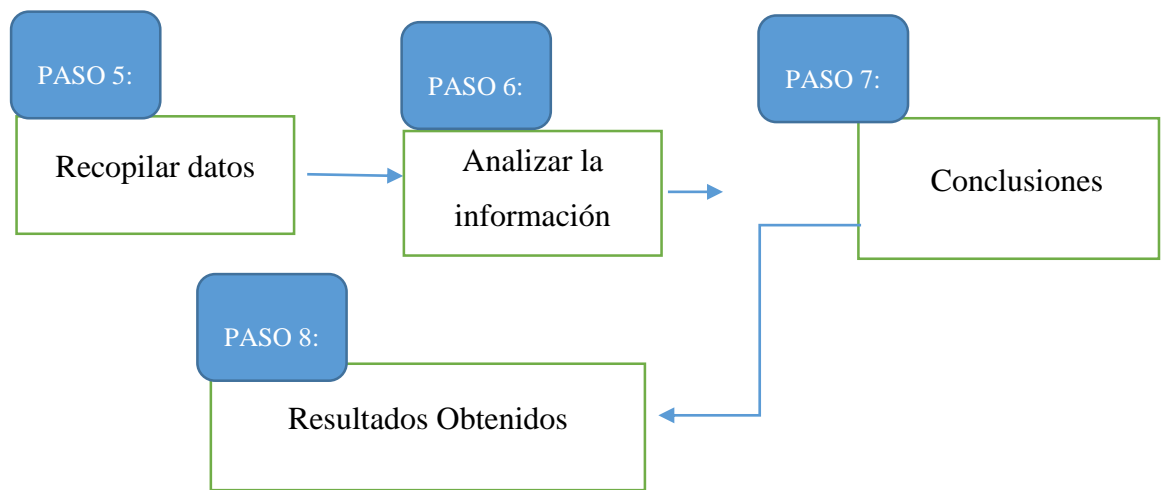


Elaborado por: Jean Carlos Buestán Vera.

Mediante el modelo del patrón lineal de investigación cuantitativa propuesto por (Sprandley, 1980), en la Figura 3.2 se presenta la metodología para las etapas en el desarrollo para la investigación propuesta.

Figura 3.2 Patrón Lineal de la investigación cuantitativa





Método para proceder a una investigación mediante etapas de desarrollo en la metodología para un patrón lineal cuantitativo. Elaborado por: Jean Carlos Buestán Vera.

3.1 Requerimientos para implementar la red IoT (Internet de las cosas)

Para el desarrollo de una arquitectura IoT, y que la tecnología sea viable sin interferencias, se debe permitir que la red sea distribuida para que los nodos puedan comunicarse entre ellos, así como también que sean escalables, seguros y eficientes.

A continuación, se procederá a entender las bases fundamentales de una red IoT:

a) Tecnología distribuida

- Distribuir objetos y dispositivos en nuestro entorno.
- Sistemas para acceder, enviar y leer datos.

b) Estandarizar que protocolo de comunicación

c) Arquitectura Escalable

d) Energía Renovable y Eficiente

e) Punto de acceso: Tecnología LoRa.

f) Datos: eje central basado en reglas y alertas.

Antes de proceder a la explicación en la implementación de software y hardware, se explicará cómo proceder para escoger procesos matemáticos y simulaciones previas a la misma.

3.2 Mediciones previas en el vehículo

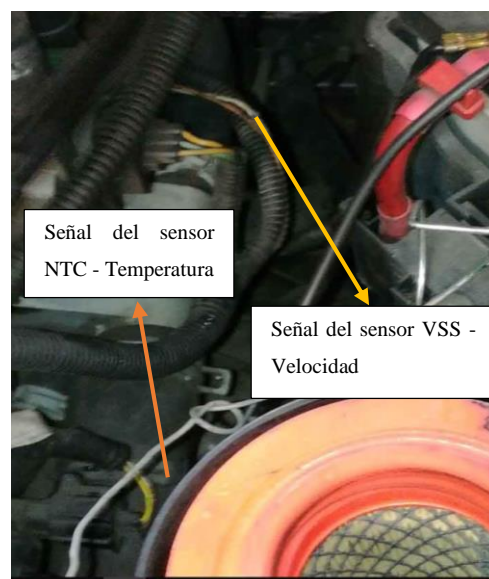
A través del vehículo modelo, se realizaron las mediciones en base a las señales que salen directamente de la ECU (Unidad de control del motor). Mediante los sensores se puede efectuar un estudio previo del vehículo. Por otra parte, gracias a un multímetro en una escala de 0 a 5 voltios, el cual trabaja cada sensor con una señal cuadrada, se

evaluaron varias mediciones a nivel de temperatura, velocidad y nivel de gasolina. Para realizar esto, cada sensor tiene una resistencia con la cual nos permite saber el estado propio al medir. En este caso, el sensor de temperatura es un ECT (sus siglas en inglés son Engine Coolan Temperature). También existe un uno del tipo NTC (Negative Temperature Coefficient), cuya resistencia cambia con respecto a la temperatura del refrigerante del motor a través de otra; es decir, entre más caliente el sensor, menor en su resistencia.

El sensor de velocidad se reconoce como CKP (sensor de posición del cigüeñal) o VSS (Vehicle Speed Sensor), el cual medirá el número y las ranuras en el plato de levas. Al hablar del medidor de gasolina hay que entender que es un REOSTATO, es decir, una resistencia variable que varía el voltaje según como esté el tanque del vehículo: si el tanque está lleno, el medidor marca baja resistencia mientras que si el tanque está vacío el medidor marca alta resistencia.

A continuación, se observará cómo se realizaron las medidas en voltaje y resistencia de los sensores a supervisar.

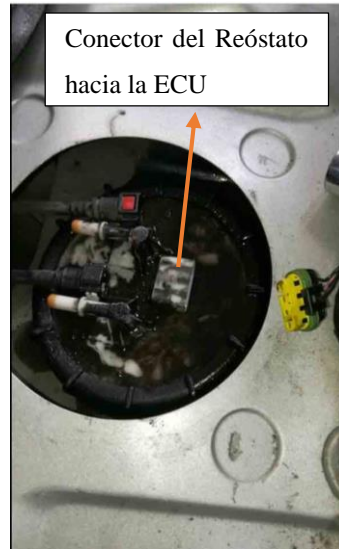
Figura 3.3. Adquisición de valores en voltaje de los sensores de temperatura y Velocidad



El sensor de temperatura del vehículo escogido, la medida de la temperatura se centró en el refrigerante y está ubicado en la parte delantera del motor y en el caso de la velocidad el sensor está ubicado por detrás de las bujías. Elaborado por: Jean Carlos Buestan Vera

Para adquirir los valores de la gasolina, se abrió el tanque desde la parte superior y se substrajo la unidad de medida. En este caso, el Reóstato es similar a un medidor tipo boya. Se tomó los valores con el multímetro simulando los 3 estados: vacío, medio y lleno.

Figura 3.4. Extracción del Reóstato para medir el nivel de Gasolina



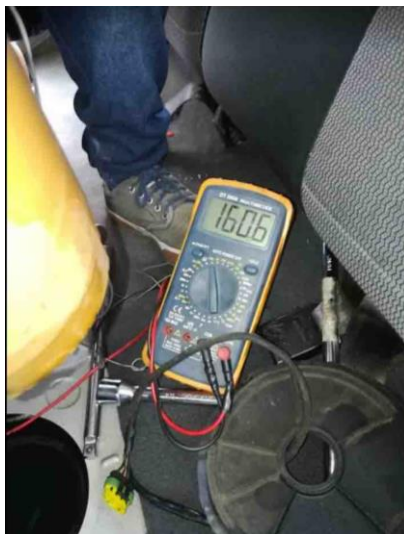
Para no desarmar, donde se deposita la gasolina se procedió a desarmar unidad de medida que es el Reóstato ubicado en la parte de debajo de los asientos de los acompañantes y sacar el sensor tipo boya para tomar las medidas. Elaborado Por: Jean Carlos Buestan Vera

Figura 3.5. Reóstato extraído del depósito de la gasolina.



Sensor tipo boya conocido como Reóstato extraído del vehículo elegido que en este caso se toma las medidas en voltaje. Elaborado Por: Jean Carlos Buestan Vera

Figura 3.6. Valor tomado de Reóstato en ohmios



Valor tomado cuando el tanque se encuentra lleno en una medida de ohmios. A través de los valores adquiridos por el multímetro, se procedió a medir el voltaje en las diferentes posiciones en las que se encuentra la boya. En este caso cuando está vacío, lleno y en la mitad la gasolina. Elaborado Por: Jean Carlos Buestan Vera

Tabla 3.1. Valores tomados de la gasolina en Voltaje

Nivel Porcentaje %	Voltaje (V)
Máximo - 100	4.98
75	3.80
Medio - 50	2.63
25	1.56
Vacío - 0	0.49

Valores tomados a una escala de 0 a 5 Voltios del nivel de gasolina. Elaborado por: Jean Carlos Buestan Vera

En lo que se refiere a las medidas tomadas a nivel de velocidad, se analizó el sensor a nivel de frecuencia y voltaje.

Tabla 3.2. Análisis de la Velocidad y Amplitud tomada con un Scanner Automotriz

Velocidad (Km)	Voltaje por División (V)
0	1
5	5
10	7.8125
20	15.625
40	31.25
60	46.875
80	62.5
100	78.125
120	93.75

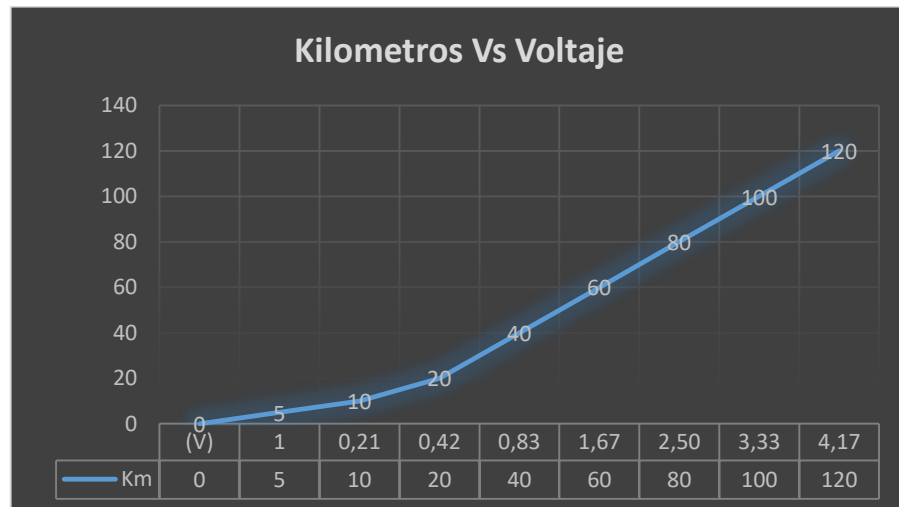
Los valores tomados están evaluados por un Scanner automotriz con un tiempo de 2 milisegundos y cada división que se tenga en voltaje se verá con el aumento de la velocidad. A mayor velocidad mayor frecuencia. Elaborado Por: Jean Carlos Buestan Vera

Figura 3.7. Velocidad evaluada con Scanner automotriz



En el Scanner se puede visualizar que el tiempo evaluado es de 2 milisegundos con 50 voltios por división. Además a una velocidad de 40 MPH/64Km con una frecuencia de 1.099KHz. Elaborado Por: Jean Carlos Buestan Vera

Figura 3.8. Relación Kilómetros Vs Voltaje



La relación que se presenta en el esquema está en una escala de 5 voltios. Cada sensor emite una señal a la escala mencionada y así poder tener la opción de trabajar con los equipos de LoRa y ARDUINO. Elaborado Por: Jean Carlos Buestan Vera

Con respecto a la temperatura antes mencionada, hay que tener en cuenta que cuando el automóvil se encuentra apagado más alta es la resistencia, mientras que cuando la temperatura es máxima, ésta disminuye.

Figura 3.9. Medida del valor de la resistencia del sensor de temperatura ECT tipo NTC.

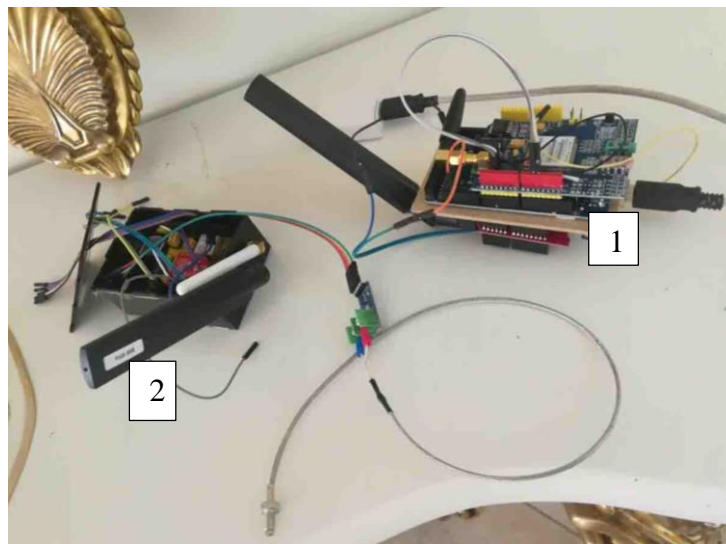


Medida del valor del sensor a nivel de refrigerante y Termostato. Elaborado Por: Jean Carlos Buestan Vera

En lo que se refiere a la implementación y comunicación segura con la WEB, se procede a realizar dos tipos de enlaces, el uno consta de LoRa Shield GPS y el otro un repetidor que cumple la función de receptor los datos que adquiere a través de los sensores implementados en el vehículo.

En lo que se refiere a la parte del uso de LoRa Shield, el cual servirá como un repetidor para expandir la señal, cuenta con el Arduino MEGA por temas de memoria que necesita para adquirir los datos y procesarlos, para lo que es el Shield este módulo trabaja con el Arduino Uno. Además lo que compone este enlace, también consta de un GPRS SIM 900 con un chip de CLARO para redirigir la información hacia la página WEB y mostrar todos los datos Adquiridos.

Figura 3.10 Módulo Shield LoRa GPS y Módulo Repetidor Shield de LoRa – GPRS



En la figura 3.21 se observa los dos tipos de enlaces desarrollados para la adquisición de datos en el vehículo como la conexiones entre arduino UNO con el MEGA, en lo que se refiere a los números el numero 1 señala al módulo repetidor con el GPRS y el numero 2 señala al Shield LoRA GPS.

En el módulo GPS de LoRA van los sensores del automotor, en el número 2 que señala dicho enlace tenemos el sensor de temperatura que es una termocupla tipo tornillo JK con recubrimiento metálico. También tenemos las antenas que son de 868 MHz, las cuales permiten tener un enlace correcto al momento de pedir la información y tener la respuesta de cada módulo.

Para adquirir los datos reales del vehículo se procedió a implementar los sensores y adquirir la señal propia que viene de la ECU con respecto al nivel actual de la gasolina, la señal como se mencionó viene del tanque que contiene un reóstato con el cual se obtiene la medida real del combustible.

Figura 3.11 Sensores en el vehículo.

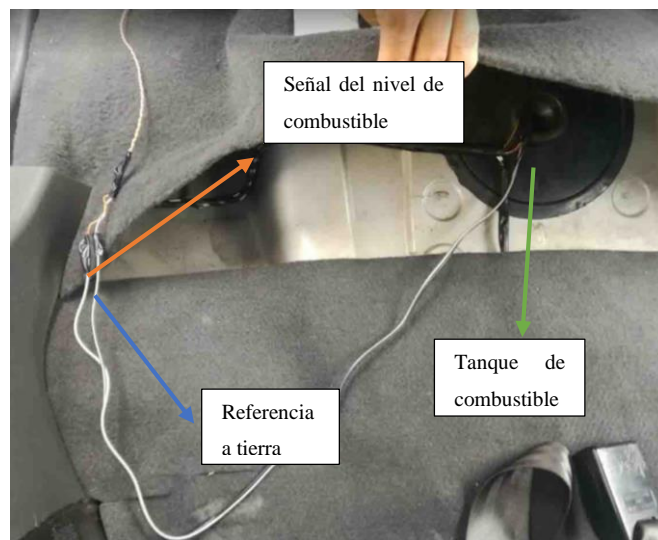


En la medida del nivel de agua se implementó un sensor ultrasónico que tiene por el momento 10 cm de alcance según la medida del tanque, en lo que respecta a la temperatura se optó por una termocupla con un recubrimiento metálico a nivel del motor.

Los dos sensores que se muestra en la figura 3.22 van conectados al módulo Shield LoRa GPS que receptara los datos que tenga el vehículo para enviarlos al repetidor que consta del GPRS que será el medio por el cual se suba la información directamente a la WEB.

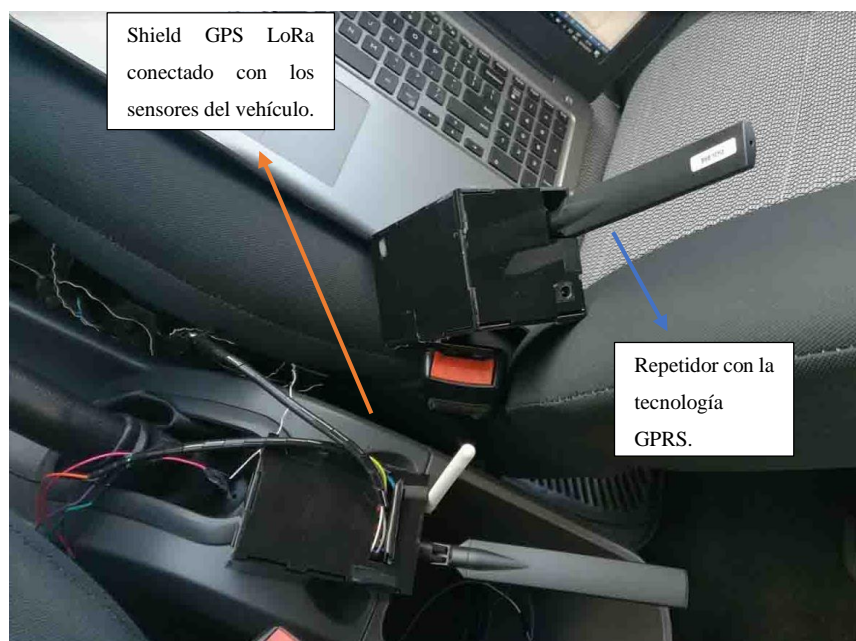
A nivel de la medida de gasolina se procede a desmontar el tanque del vehículo donde se ubica el combustible para extraer el aforador o reóstato el cual es un mecanismo tipo boya que mide el nivel de líquido como una resistencia. A través del conector que cuenta este medidor de combustible, se encontró la señal que brinda la medida del nivel de líquido para proporcionar el dato al módulo GPS de LoRa.

Figura 3.12 Señal adquirida desde el tanque del vehículo.



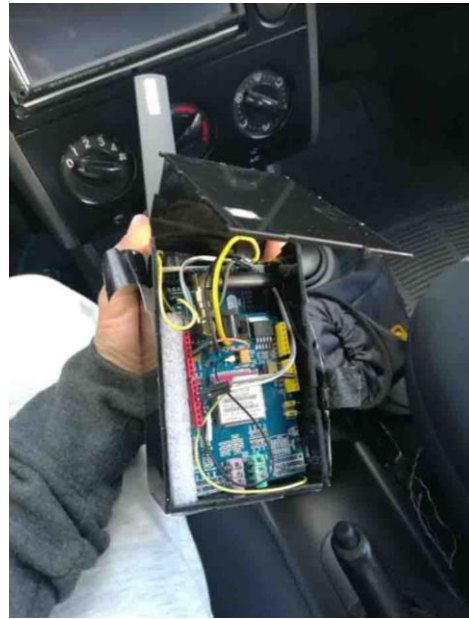
A través de la extracción de la señal del aforador que consta de un conector que va directo al tablero del vehículo para observar el nivel de combustible, se adquirió la medida real para proporcionar el dato al módulo de Shield GPS LoRA que está en el vehículo para enviar al módulo receptor.

Figura 3.13 Módulo Shield GPS LoRA en el vehículo.



En la Figura 3.13 tenemos el módulo que permite recibir los parámetros medidos en el vehículo y así poder enviar al módulo repetidor que cuenta con la tarjeta GPRS SIM 900 Para enviar la información a la WEB.

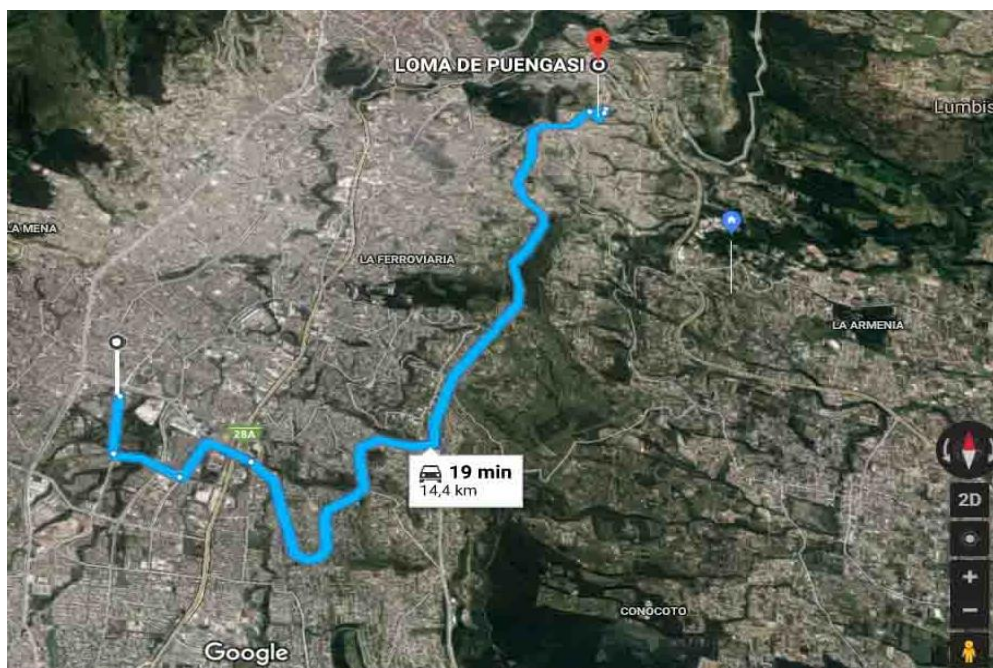
Figura 3.14 Repetidor con tecnología GSM- GPRS SIM 900



Este dispositivo consta de dos Arduinos, el principal es para el Shield de LoRA que receptara los datos del módulo montado en el automotor y el otro para recibir los datos para subirlos mediante la tecnología GSM.

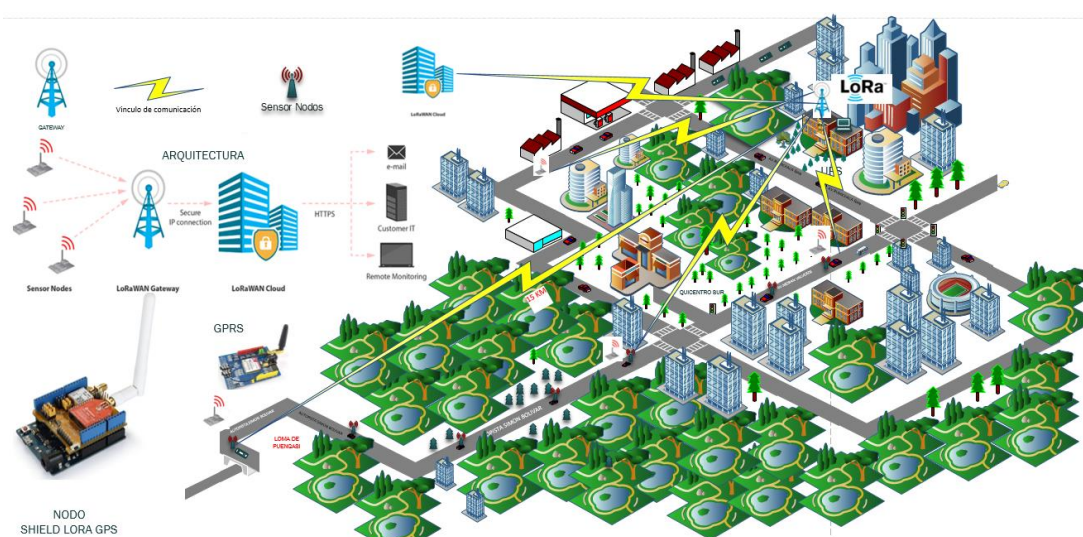
3.3 Explicación del tramo elegido y Aplicación de Modelo Matemático Okumura-Hata para el área suburbana

Figura 3.15 Trayectoria Optada para realizar la implementación de la Red.



Trayectoria es la que se va a tomar al momento de transmitir los datos mediante conexión RF (radio Frecuencia) con los equipos LoRa. Elaborado Por: Jean Carlos Buestan Vera

Figura 3.16 Trayectoria simulada según la arquitectura propuesta por LoRaWAN.



Simulación realizada en Microsoft Visio para observar el desarrollo de la arquitectura desde el punto de inicio y final del tramo elegido. Elaborado Por: Jean Carlos Buestan Vera

Con el fin de obtener el comportamiento, en base a las siguientes formulas, y con medidas aleatorias y medidas para relaciones numéricas, se manifestara el manejo del modelo para obtener los factores de corrección con el tipo de antena y altura. Esto depende de la zona con la que se enfrentan las frecuencias al momento de elaborar una red, en este caso mediante la atenuación media en espacios libres.

$$Lb = 69,55 + 26,16 \log f - 13,82 \log hb - a(hm) \\ + (44,99 - 6,55 \log h b) \log dm$$

Expresa la perdida básica mediante el modelo de Okumura-Hata

Donde $a(hm)$ = factor de corrección que depende del tipo de antena y altura.

Áreas Rurales

$$Lb = Lb(urbano) - 4,78 \log f^2 + 18,33 \log f - 40,94$$

Factor de corrección rural

- Áreas Urbanas

Ciudades pequeñas o medianas ($1 \leq hm \leq 10$ m)

$$a(hm) = (1,1 \log f - 0,7) hm - (1,56 \log f - 0,8)$$

Factor de correccion en ciudades pequeñas

- Ciudades Grandes:

Figura 3.17 Factor de corrección ciudades grandes (Garces, 2017)

$$a(h_m) = \begin{cases} 8,29(\log 1,54h_m)^2 - 1,1, & f \leq 200 \text{ MHz} \\ 3,2(\log 11,75h_m)^2 - 4,97, & f \geq 400 \text{ MHz} \end{cases}$$

A través de un modelo matemático, nos centraremos en obtener la pérdida básica de propagación en función de las alturas de las antenas de la estación base y móvil. Al mismo tiempo, también la distancia como la frecuencia.

El método de cálculo proporciona buenos resultados para los entornos suburbanos y urbanos en los cuales se desarrollaron las pruebas de conectividad con el vehículo. El modelo que brinda el cálculo es el de Okumura-Hata. A su vez, se realizara el cálculo de corrección por zona de recepción para el área suburbana en donde se desarrolló el tramo elegido.

Perdida básica de propagación

$$\begin{aligned} Lbu &= 69.55 + 26.16 \log(f) - 13.82 \log(ht) - a(hm) \\ &\quad + (44.9 - 6.55 \log(ht)) \log(d) \\ a(hm) &= 3.2 * (\log 11.75(hm))^2 - 4.97 \end{aligned}$$

f = frecuencia portadora

ht = altura estación base

hm = altura antena móvil

d= distancia entre la estación base y terminal (km)

$$\begin{aligned} a(hm) &= 3.2 * (\log 11.75(10))^2 - 4.97 \\ a(hm) &= 8.74 \\ Lbu &= 69.55 + 26.16 \log(868\text{Mhz}) - 13.82 \log(10) - 8.74 \\ &\quad + (44.9 - 6.55 \log(10)) \log(10\text{Km}) \\ Lbu &= 434.22 \end{aligned}$$

Zona Suburbana: corrección por zona de recepción

$$Lb = Lb - 2(\log \frac{f}{28})^2 - 5.4$$

$$Lb = 434.22 - 2\left(\log \frac{868\text{MHz}}{28}\right)^2 - 5.4$$

$$Lb = 318.58 \text{ dB}$$

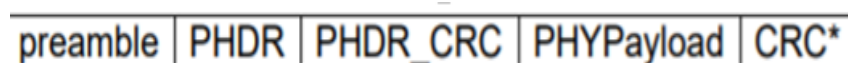
Como resultado, el modelo me abre la probabilidad de una señal que llegue o no a su destino desde el transmisor al receptor. Dicho resultado de la zona suburbana predijo la pérdida en dB de una potencia dentro de la zona mencionada.

3.4 Formato de la Trama Física de LoRa

Los paquetes siguen una estructura definida por el protocolo LoRAWAN, ya que cuentan:

- Preámbulo de 8 bytes de longitud
- Cabecera (PHDR)
- Payload
- CRC.

Figura 3.18 Formato de la trama física (**Ordoñez, 2017**)



Elaborado por: Jean Carlos Buestan Vera Por: Jean Carlos Buestán Vera.

El preámbulo permite definir parámetros de forma individual para cada paquete. Este indica el instante para transmitir, así como también el esquema en función de distancia y el consumo energético que es importante en la implementación de una IoT. La cabecera está definida por un solo byte en la trama (PHDR).

Payload tiene un tamaño máximo de 255 bytes y contiene la capa de acceso al medio de Lora WAN. La cabecera MAC identifica el tipo de mensaje y el formato de la trama dentro de la capa LoRaWan con la que ha sido asociada. A su vez, existen varios tipos de mensajes MAC, descritos en la siguiente tabla:

Tabla 3.3 Tipos de mensajes MAC (Garcia, 2017)

TIPO	DESCRIPCION
000	Join Request
001	Join Accept
010	Unconfirmed Data Up
011	Unconfirmed Data Down
100	Confirmed Data U
101	Confirmed Data Down
110	RFU
111	Proprietary

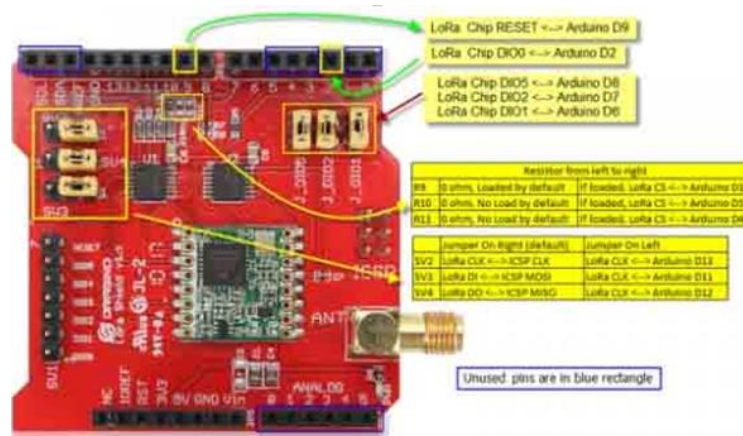
MACPayload: tiene una cabecera en la trama, puerto opcional y un campo de trama opcional.

Elaborado Por: Jean Carlos Buestan Vera

Dentro del desarrollo de la tecnología LoRa, nos afianzamos en los productos de Dragino en lo que se refiere a Hardware para esta tecnología. Al mismo tiempo, también en productos propios de SEMTECH y Microchip, las cuales son marcas dedicadas a la tecnología mencionada. Así pues, antes de empezar a manejar las placas de ARDUINO, Shields de Dragino y nodos LoRA, se dará a conocer parámetros importantes de los equipos a usar.

A continuación veremos el correcto uso de las placas de Dragino shield, Modulo GPS y esquemas explicativos de los pines a usar:

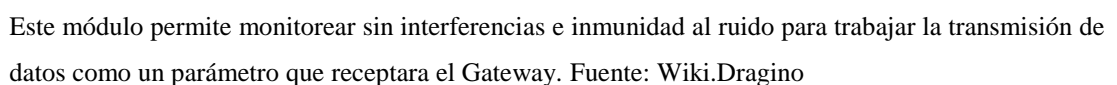
Figura 3.19 Dragino Shield con pines para conexión con Arduino



En esta figura se puede observar los pines que se usan para la comunicación entre lora y ARDUINO, D13, D12, D11 y Cs, Chip Select que puede ir en D10, D5 o D4, es decir se usa comunicación ICSP entre Shield y el ARDUINO. Fuente APRENDIENDO ARDUINO.

En lo referente a uno de los parámetros más significativos en los que se fundamenta el estudio de la tecnología LoRa, éste es el monitoreo del vehículo a una gran distancia mediante el uso del Gateway, el cual se conectara a Internet con varios sensores y brindará los datos supervisados. Mediante un conjunto de componentes que conforma Dragino LoRA/GPS Shield y Dragino LoRa Bee V1.0 con una frecuencia de 868 MHz, tenemos la posibilidad, con ARDUINO, de tener como tal un módulo GPS a larga distancia.

Figura3.20 Dragino LoRA/GPS Shield con pines selectivos para Arduino.

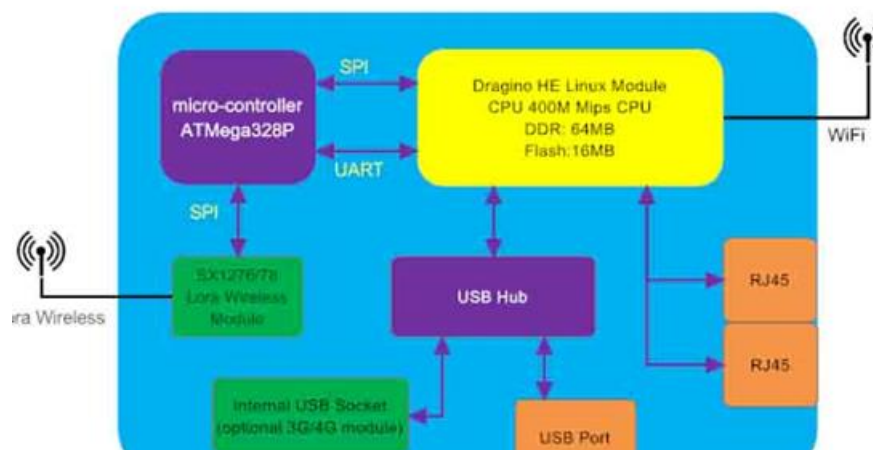


Para empezar a realizar las pruebas de desarrollo, me basaré en LoRa IoT development Kit, el cual me proporciona las herramientas iniciales para desenvolver el manejo de los módulos Shield de LoRa con distintos sensores y placas de Arduino. Esto hará posible implementar la red IoT en base a los parámetros antes mencionados para supervisar dentro del automóvil.

Con el fin de conectar a internet los sensores, el Gateway LG01 LoRA brindara la conexión para enlazar los parámetros medidos en el vehículo. Esto hará posible tener el manejo con el control mediante la recepción de dichos datos, lo cual estará dentro de la arquitectura de red que recibe en el nodo final los datos transmitidos, y a su vez reenvía los paquetes al servidor centralizado.

Figura 3.21 Arquitectura Gateway LG01

LG01 System Overview:



Mediante esta arquitectura tenemos a la vista el módulo LoRa SX1276/78 y un microcontrolador que me permitirá trabajar en forma bidireccional hacia el módulo Dragino de Linux. Fuente: Aprendiendoarduino

Después de conocer las características, arquitectura y sistema de Hardware, procedemos a realizar la detección con la red WiFi que se encuentra con el nombre dragino-168cb0. Luego de conectar la red como tal mediante un navegador, procedemos a ingresar la IP: 10.130.1.1, lo cual me permitirá ingresar a la interfaz de DRAGINO para escribir el usuario y la contraseña como tal, y así desplegar una pantalla con las especificaciones del Gateway.

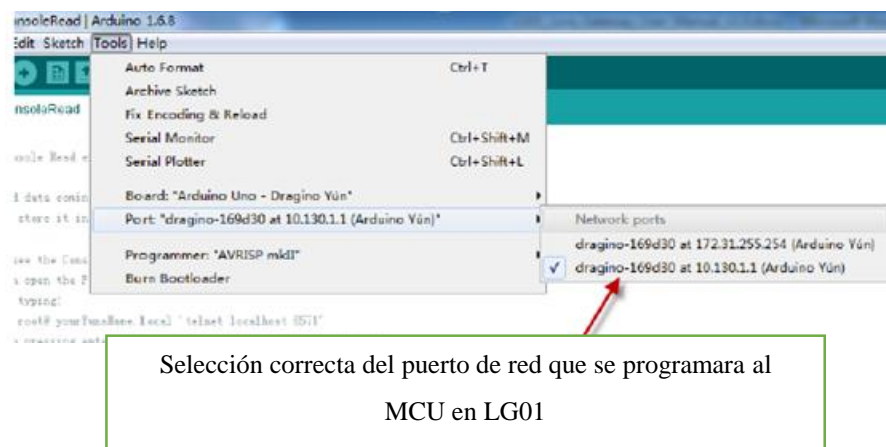
Figura 3.22 Interfaz de acceso al conectar con la dirección IP 10.130.1.1



En lo que se refiere a la contraseña y usuario hay que tener en cuenta que siempre tendrá relación a DRAGINO ya que ellos desarrollaron el acceso a LG01 con WiFi. Elaborado por: Jean Carlos Buestán Vera

Mediante la plataforma de Arduino tenemos la opción de descargar la librería de Dragino Yun, que permitirá conectar con la placa Shield de LoRa o conjuntamente con el Gateway que principalmente ofrece la conectividad entre dispositivos, para conectar específicamente directamente con LG01 se tiene que escoger con la misma dirección IP antes mencionada 10.130.1.1 cómo se mostrara a continuación:

Figura 3.23 Selección de puerto Dragino-169d30 sobre la IP: 10.130.1.1 en Arduino Yun



Con este procedimiento garantizamos que cualquier programación desarrollada en ARDUINO se conecte y suba el Sketch de forma correcta para enlazar la comunicación con las placas Shield de LoRa y evitar mensajes de error al subir a la placa la información programada como tal para desarrollar conectividad con el Gateway. Elaborado por: Jean Carlos Buestan Vera

A través de las recomendaciones dadas se procedió a interactuar el envío de datos como Cliente mediante LoRa Wireless con los siguientes componentes los cuales permitirá tener acuse de recibo:

- LoRa Gateway: LG01

LoRa Cliente: LoRa Shield + Arduino Uno

Figura 3.24 Dispositivos LoRa para realizar comunicación entre cliente y servidor



Evaluación con los equipos para enlazar y probar los enlaces tanto del Sketch del servidor como del cliente. Elaborado por: Jean Carlos Buestan Vera.

Figura 3.25 Comunicación LoRA en el Gateway LG01.



Como se observa al momento de tener comunicación se enciende el led de LoRa para indicar que está transmitiendo de forma continua el mensaje. Elaborado por: Jean Carlos Buestan Vera

3.5 Registro de datos mediante una doble cámara incorporada en el vehículo

Mediante una doble cámara incorporada en el vehículo, se procede a verificar los aspectos de manejo del usuario y observar la velocidad, posición GPS satelital y las coordenadas donde se encuentra el usuario al transitar por el tramo elegido (autopista). El dispositivo instalado, aparte de los sensores a supervisar, pretende dar una evidencia en tiempo real grabando el entorno interior y exterior del automotor para tener evidencia de donde se encuentra, y por donde conduce el usuario en caso de presentarse desperfectos del automotor.

Al mismo tiempo, mediante la cámara al exterior se quiere comprobar si el tramo elegido con la hora y fecha presenta tráfico en diferentes horarios, tomando la misma ruta, ya sea de ida y retorno del vehículo. El dispositivo cuenta con un sensor G y GPS satelital, además de un velocímetro para constatar los kilómetros al transitar por la autopista. Brinda un software conocido como JPlayer, que permite desplegar la información adquirida a través de una tarjeta SD que guarda los parámetros medidos al conducir.

Figura 3.26. Verificación mediante imágenes del trayecto elegido con los parámetros de velocidad, GPS satelital y coordenadas.



A través del dispositivo incorporado al vehículo se puede constatar el tramo elegido con sus respectivas mediciones, como son la velocidad que se encuentra en 83.4 Km/h, posición satelital, coordenadas y monitoreo de inclinación del automotor mediante el sensor G. Elaborado por: Jean Carlos Buestan Vera

CAPÍTULO 4

PRUEBAS Y RESULTADOS

Para lograr una red IoT se necesita de un medio por el cual los datos adquiridos se deben manejar y registrar en la nube, o como tal a través del internet de las cosas. Esto se puede conseguir mediante una plataforma que nos brinda un servidor orientado a las aplicaciones desarrolladas en conjunto con ARDUINO y sobretodo en conjunto con las placas de Dragino que brindaran los datos de posición, temperatura y niveles de líquidos en el vehículo.

4.1 Manejo de la plataforma ThingSpeak

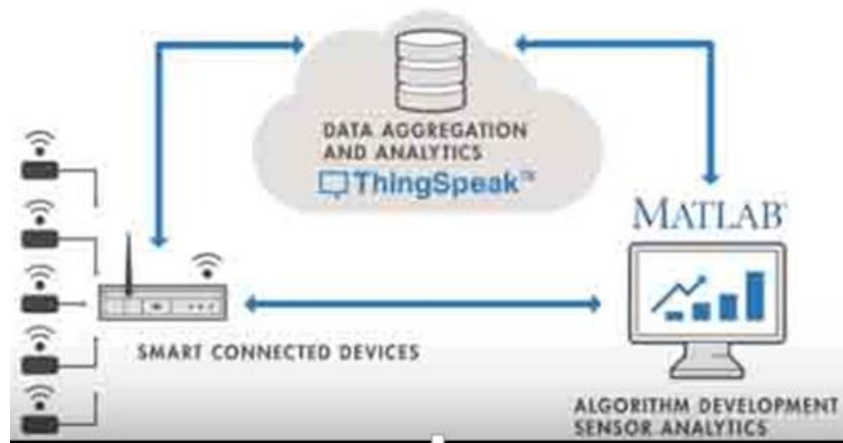
La aplicación ThingSpeak maneja la adquisición de los parámetros a medir mediante canales, consta de 8 campos de datos, campos de ubicación y de estado. Para nuestros fines, mediante estos canales será posible recolectar la información necesaria de cada sensor ubicado en el vehículo y almacenar dichos datos en esta plataforma. Posteriormente, esto se podrá representar visualmente para poder mostrar los datos en pocos instantes, ya que cada 15 segundos se recopila cada parámetro medido. La aplicación mencionada consta de una API que permite transmitir y almacenar los datos en el protocolo HTTP, y al ser un código abierto, puede ser público y privado ya que se pueden crear y eliminar los datos adquiridos. A su vez, al pertenecer a Mathworks, podemos trabajar con Matlab y simular mediante esta plataforma los datos visualmente para ser manejados por el internet de las cosas.

Figura 4.1 Interfaz de la plataforma ThingSpeak.



Mediante esta plataforma se crearán los canales necesarios para recopilar los datos de los sensores y almacenarlos visualmente en el internet de las cosas. Elaborado por: Jean Carlos Buestan Vera.

Figura 4.2 Arquitectura para una red IoT con la plataforma ThingSpeak.



A través de esta arquitectura entendemos el desarrollo del proceso y manejo de los datos visualmente mediante los parámetros que nos brinda los sensores con las placas de LoRa Shield. Fuente: ThingSpeak.

Para realizar el canal respectivo, en este caso solo se necesitará crear uno solo, por el hecho de que nuestra red está conformada por un solo nodo que nos brindará la lectura de varios sensores dentro del automóvil. En este caso se crearon 6 campos para registrar las lecturas respectivas: Latitud, Longitud, Velocidad, Temperatura, Nivel de Agua y Nivel de gasolina. Para cada canal se tiene una identificación específica que nos servirá para enlazar el reconocimiento con la placa de ARDUINO y a su vez también tiene una llave de API conocido como Write API key, la cual consta de 16 dígitos que permite una escritura de datos en un canal. Para leer la clave API, esta permite leer los datos almacenados en un canal. La API (por sus siglas en inglés Application Programming Interface) se puede definir como “interfaz de programación de aplicaciones” (INTERNETYA, 2018). Esta se presenta como una solución de alta calidad de software que nos lleva a tener la seguridad en los dispositivos conectados a cliente, lo cual es fundamental ya que es la llave de acceso al mundo de la IoT.

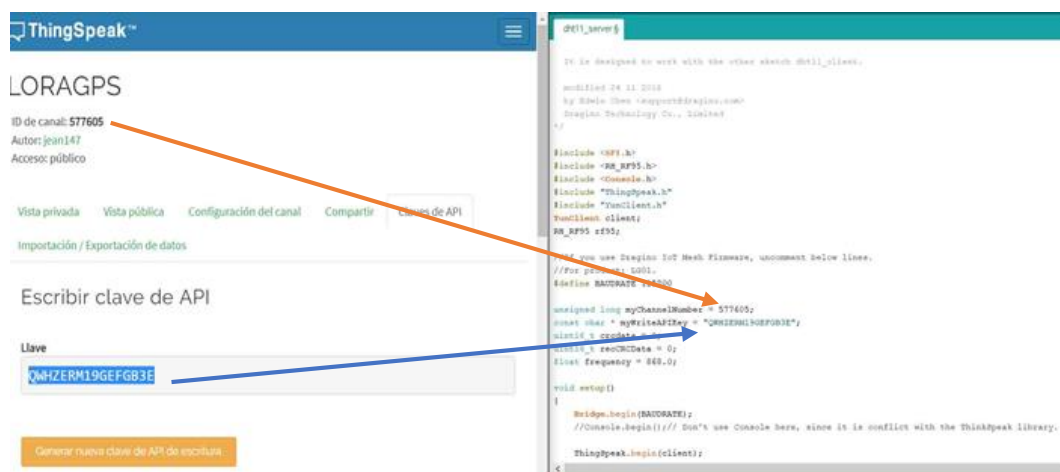
También se debe tener en cuenta que, para lograr tener el internet de las cosas, es necesario una API REST para cada campo (Alexandra, 2017), ya que estos interactúan con los protocolos de internet y sobre todo para tener una verdadera conectividad M2M (Maquina a Maquina). En tal sentido, el éxito es interactuar con el mundo físico mediante la construcción de aplicaciones a través de estas herramientas.

Figura 4.3 Configuración del canal de ThingSpeak con 6 campos de lectura

Al momento de crear un canal se crea una identificación y una llave API para poder tener enlace con la programación de ARDUINO. Elaborado por: Jean Carlos Buestan Vera

A continuación, se mostrará donde se debe ubicar la identificación del canal y la llave de escritura API en la sección de programación de ARDUINO como tal.

Figura 4.4 Configuración entre ThingSpeak y ARDUINO mediante el canal de identificación



Mediante este proceso puedo identificar el parámetro del servidor para poder registrar visualmente el comportamiento de cada sensor o campo representativo en la interfaz. Elaborado por: Jean Carlos Buestan Vera

ThingSpeak brinda la opción mediante un procedimiento en base a histogramas para observar el comportamiento de los sensores en un procesamiento de información para que la interacción dinámica en la plataforma se comuniqué de forma inter e intra vehicular. Se pensó en una mejor interacción con el usuario y manejo de los datos mediante reportes, ubicación exacta del objeto (vehículo), aspectos básicos que necesita tener en cuenta el conductor al transitar, una plataforma en base a un servicio WEB.

4.2 Funcionamiento de la Pagina WEB

Para poder efectivizar todo este procedimiento se adquirieron un dominio y un Hosting (almacenamiento) gratuitos, a través de noip.com (capacidad de hasta 3 dominios). El dominio adquirido se representa como loraups.ddns.net; el hosting está desarrollado en Linux Ubuntu. Para realizar el ambiente virtual a través del internet, se envuelve en ambientes de programación como PHP, CSS, JavaScript, HTML y SQL:

- HTML: proporciona la forma a la página web como encabezado y cuerpo.
- PHP: conexión para la base de datos para grabar y extraerlos.
- CCS: permite dar colores y tipo de letra.
- JavaScript: permite que sea dinámico en las actualizaciones para ser automáticas y para la representación visual se lo realiza mediante Framework llamado Highcharts.

La Programación para cada campo está explicado y se evidenciara en los ANEXOS.

En este sentido se considera que la página web resulta ser un aporte extra para mantener una alternativa de ofrecer una mejor interacción entre aplicación-usuario. La página WEB trabaja con un servidor propio llamado SQL base de datos creado por campos en (MYSQL) la cual crea los campos de la base de datos, los valores a subir son flotantes y en los rangos a trabajar mediante esta página no existen valores máximos ya que son auto escalables. El rango de los valores a graficar al momento de receptar los valores de los sensores varía entre 1000 a -735.40, los cuales son enteros y decimales.

La página consta con un registro de hora y fecha en la cual se actualiza la recepción de datos. Además, se visualizará un espectro de gráficas, en una sola los valores adquiridos y de forma independiente cada una para observar con detalle cada sensor. La página también dispone de una opción que es ver un mapa, en el cual se interpretará la longitud y latitud de la ubicación del vehículo y se mostrará en google Maps. Al obtener los valores y al ser graficados también proporcionan un reporte PDF que desplegara los datos mediante una tabla los datos receptados.

Figura 4.5 Interface de la Pagina WEB.



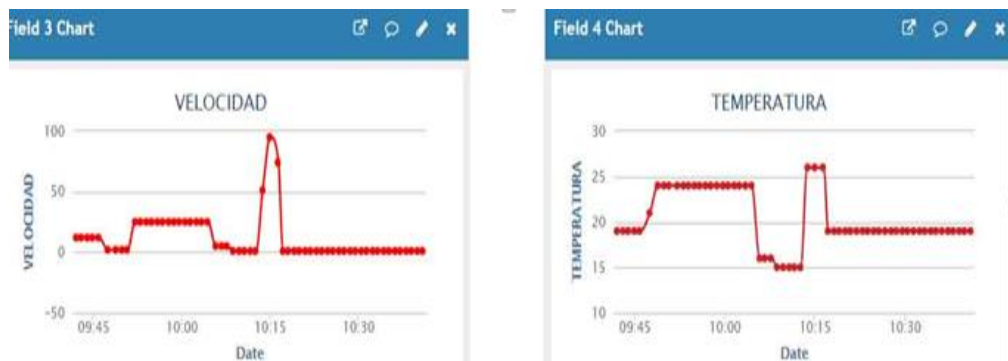
Dentro de la plataforma de loraups página web, se tiene la interface que permite al usuario observar de manera interactiva un reporte detallado de cada parámetro según la fecha y hora que se adquirió el dato medido. Elaborado por: Jean Carlos Buetán Vera.

La página Web tiene la opción de tener una representación gráfica general e individual de cada parámetro medido que en este caso será la temperatura, Nivel de Agua, Nivel de gasolina y Velocidad del vehículo, además consta de la adquisición de Latitud y Longitud que brinda la opción de dar Click en la opción ver mapa que direcciona abriendo una pestaña adicional hacia GOOGLE MAPS la cual dará la ubicación exacta del vehículo monitoreado.

4.3 Representación de los datos adquiridos en las plataformas Usadas

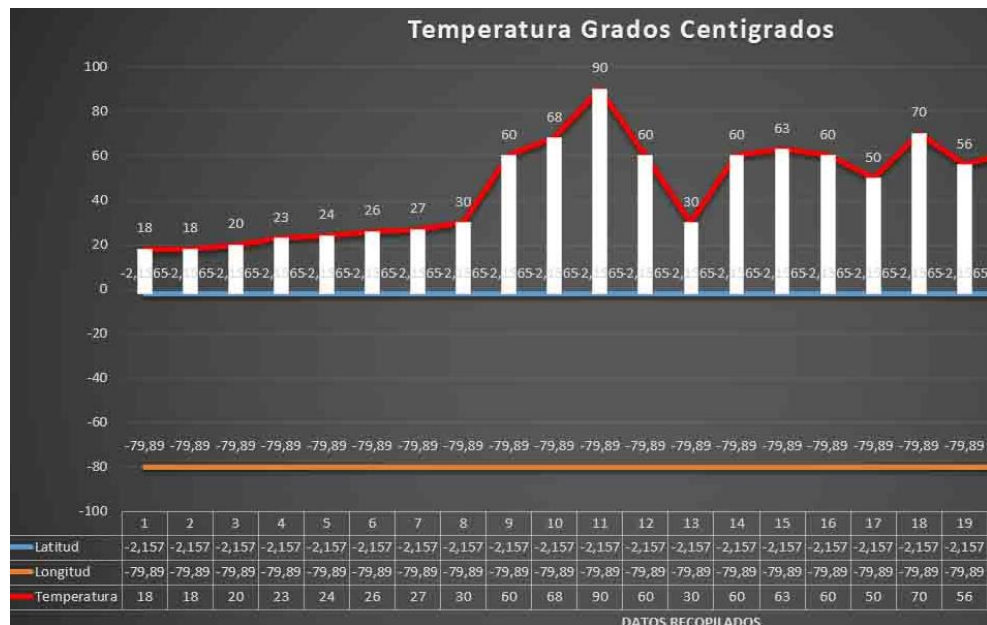
Los parámetros adquiridos mediante las diferentes plataformas se representarán mediante reportes, histogramas, interfaces interactivas, tablas y conexiones directas a otras aplicaciones dedicadas al usuario. Mediante la adquisición de datos en la plataforma propia de la IoT ThingSpeak, desarrollado en un entorno visual, podremos evidenciarlo para registrar los datos dentro del ambiente con la conexión propia entre ARDUINO con LORA/GPS.

Figura 4.6 Parametros adquiridos en ThingSpeak.



Dentro de la representación de cada parámetro se evidencia la hora que el nodo recepto el valor y sus valores en una escala de 0 a 100 la representación con los parámetros de latitud y longitud evidencia la posición GPS del vehículo. Elaborado por: Jean Carlos Buestan Vera.

Figura 4.7. Representación de los parámetros adquiridos en base a la temperatura, latitud y longitud.



Mediante una grafico resumimos el comportamiento de los parámetros adquiridos dentro del tramo elegido a través de la plataforma ThingSpeak. Elaborado por: Jean Calos Buestan Vera.

A través del desarrollo de la página WEB para hacer más dinámica la adquisición de datos a través del internet de las cosas se procede a crear un ambiente virtual, que el usuario tendrá una supervisión del vehículo con acceso público o privado.

Figura 4.8. Acceso WEB para iniciar sesión con el usuario.

UNIVERSIDAD POLITÉCNICA
SALESIANA
ECUADOR

administrador

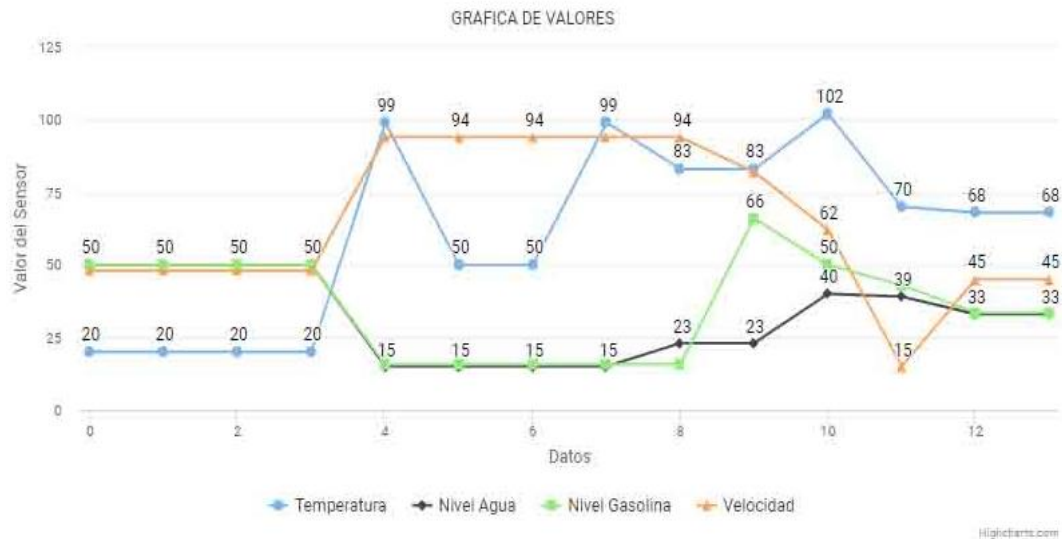
.....

Iniciar Sesión

En este acceso al usuario se pretende dar accesibilidad propia del vehículo y tener un control propio de los datos representativos de las variables que representa el estado al conducir. Elaborado por: Jean Calos Buestan Vera.

A continuación, se mostrará el ambiente representativo de los datos adquiridos:

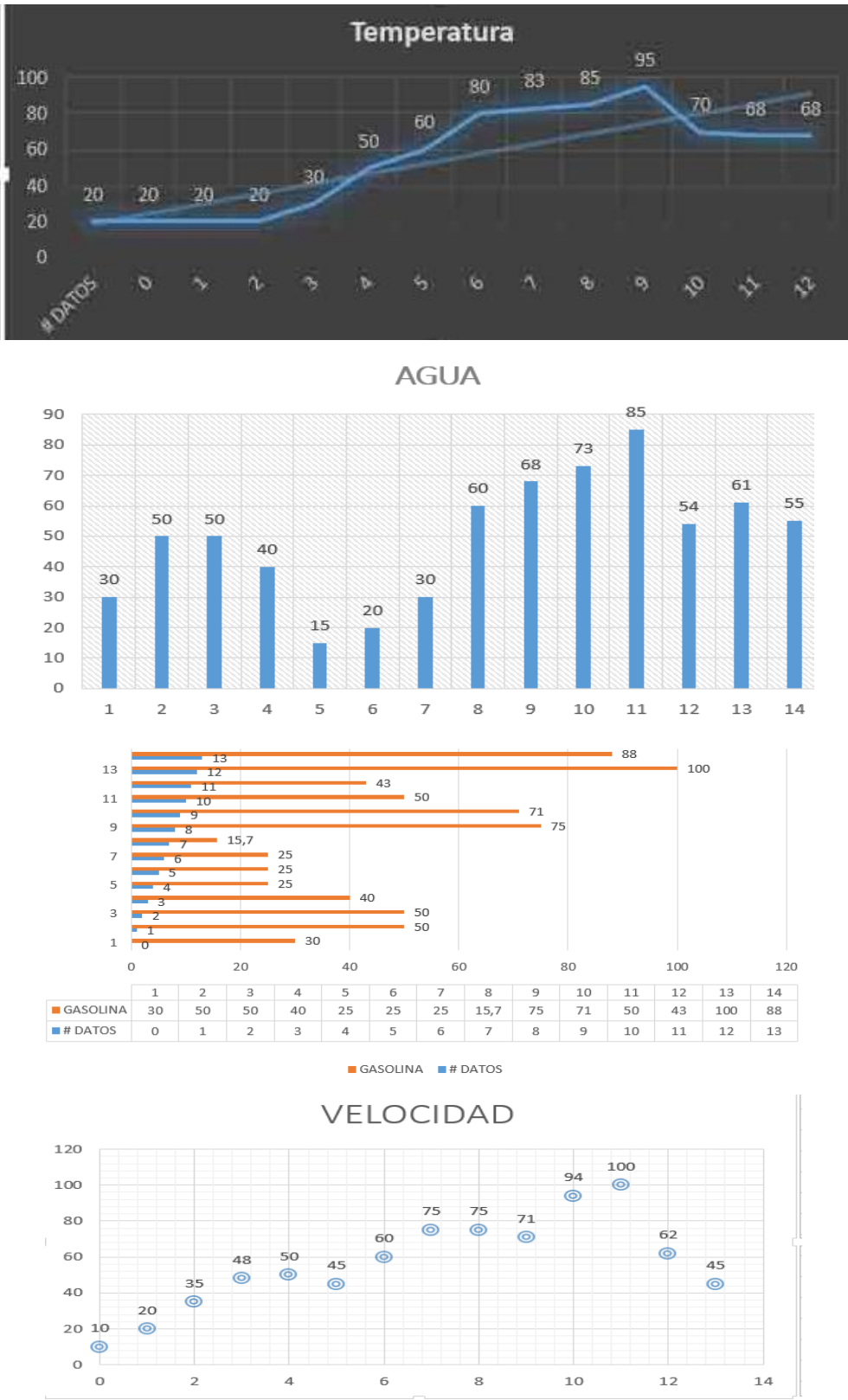
Figura 4.9. Representación de los parámetros registrados por el servidor de la WEB.



Mediante este histograma global de los datos importantes del vehículo se puede constatar que en tramos constantes los datos también lo son, según el trabajo del vehículo y trabajo que realice en la trayectoria elegida. Elaborado por: Jean Calos Buestan Vera

En la página WEB al brindar un esquema general, también realiza un histograma de cada parámetro medido individualmente para tener un estudio a nivel de que tendencia tiene el vehículo al transitar.

Figura 4.10. Representación individual de los parámetros medidos por la WEB.



Cada parámetro esta con su unidad de medida, temperatura-Grados centígrados, Agua y Gasolina-Nivel %, Velocidad-Km. Elaborado por: Jean Carlos Buestan Vera.

También tenemos la opción mediante el módulo Shield GPS Lora, a través del monitor serial, podemos desplegar mediante el software propio de arduino 1.8.5 tener la recepción de datos adquiridos por los sensores. La ventana que se desplegara se observara los datos de temperatura, nivel de agua en distancia y además los decibels con respecto al otro modulo, en si la distancia a que se encuentra interpretada en RSSI (Indicador de señal recibida).

Figura 4.12 Recepción de datos por los sensores del vehículo mediante el monitor serial.

```

tt:32
dist1:3
ag:7
rxddata
trama:T-0.310L-78.443V37T32A07G12337
trama
txLoRaServer
No reply, is LoRa server running?

-0.310430
C = 32.00

-0.310597
-0.310597
tt:32
dist1:3
ag:7
rxddata
trama:T-0.311L-78.443V41T32A07G12593
trama
txLoRaServer
got reply: RESPUESTA
RSSI: -89

-0.310807
C = 32.00

-0.310977
-0.310977
tt:32
dist1:3
ag:7
rxddata
trama:T-0.311L-78.442V43T32A07G12593
trama
txLoRaServer
got reply: RESPUESTA
RSSI: -83

```

Este mensaje quiere decir que no tuvo comunicación ni respuesta del otro módulo.

El mensaje RESPUESTA indica que el otro módulo recepto la información y se pudo subir los parámetros a la WEB.

En la Figura 4.12 tenemos la adquisición de datos con la respuesta del módulo Shield de LoRA-GPRS donde se puede observar que tt = temperatura, dist = nivel de agua, trama es la latitud y longitud del vehículo, RSSI son los decibels que tienen en la comunicación de los dispositivos al adquirir los parámetros.

A través del Arduino Mega, nos enlazamos con la tecnología GSM del GPRS SIM 900, el cual permite subir los datos de cada sensor que interviene en el vehículo, al momento que se presiona un pulsador que activa la comunicación hacia la WEB se pretende recopilar a cada minuto los datos y mostrarlos mediante el ambiente virtual loraups.ddns.net.

A continuación se mostrara el enlace y el re direccionamiento de los datos hacia la WEB para almacenarlo en el servidor SQL, Al momento de accionar la tecnología se debe reiniciar el dispositivo que se encuentra en el automotor receptando los datos de los sensores.

Figura 4.13 Monitor serial del envío de datos a través del GPRS a la WEB

```

25
BUFFER:/php/nuevo_dato.php?t=20&a=07&g=25&v=00&l=-0.310&o=-78.441&id=9***
status=ATTACHED
IP
10.211.66.247
BUFFER:/php/nuevo_dato.php?t=20&a=07&g=25&v=00&l=-0.310&o=-78.441&id=9***
Number of data received:
189
Drx:
HTTP/1.1 200 OK
Date: Wed, 12 Dec 2018 20:53:54 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.24
Content-Length: 2
Connection: close
Content-Type: text/html

okfssuSSDSSa.[fvg] (;OnwSS'SSSuSSJvS|+fwSvSSKSS/vSSSiS
T-0.310L-78.441V00T12A07G24PROCESAR
-0.310
-78.441
00
12
07
24
BUFFER:/php/nuevo_dato.php?t=12&a=07&g=24&v=00&l=-0.310&o=-78.441&id=9***
status=ATTACHED
IP
10.211.66.247
BUFFER:/php/nuevo_dato.php?t=12&a=07&g=24&v=00&l=-0.
Number of data received:
189
Drx:
HTTP/1.1 200 OK
Date: Wed, 12 Dec 2018 20:55:28 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.24
Content-Length: 2
Connection: close
Content-Type: text/html

okfssuSSDSSa.[fvg] (;OnwSS'SSSuSSJvS|+fwSvSSKSS/vSSSiS

```

IP estable de la conexión con Claro

Envío de los parámetros hacia la WEB

El estado OK nos indica que se subió con éxito a la página WEB

En la Figura 4.13 tenemos el envío de datos receptados a través del Shield GPS LoRA hacia la página WEB, en donde se direcciona a la IP de chip de claro con un BUFFER que envía los datos hacia el ambiente virtual.

Tabla 4.2. Costo de Implementación red IoT.

Descripción	Costos Materiales	Costo Software	Costo Hardware	Tiempo(Horas)	Costo Total
Camara Dvr GPS Cdr E07	\$220				\$220
LoRa IoT Development Kit	\$150				\$150
Arduino Mega	\$15,50				\$15,50
Shield módulo GSM GPRS Sim900 arduino	\$39,30				\$39,30
Sensor MAX6675 de Temperatura con Termopar Tipo K	\$10				\$10
Sensor Ultrasónico HC-sr04	\$3,00				\$3,00
Desarrollo WEB : loraups.ddns.net/login.html		\$150	\$150	1 Mes=730,001	\$300
Adquisición de la señal a nivel del tanque de Gasolina			\$15	3	\$15
Cableado de los sensores a la cabina del conductor			\$10	1	\$10
Llegada del pedido al exterior con tarifa de envío.	\$80			2 Meses = 1460	\$80
Implementación de Thing Speak con los parámetros a medir		Gratis		1 Mes=730,001	
Desarrollo de Programación en Arduino y Gateway				1 Mes=730,001	
TOTAL				3654,003	\$842,80

A través del análisis de costos se pudo determinar que el proceso de enviar los datos adquiridos por los sensores y programación toma un tiempo estimado a casi 3 meses, a su vez un tiempo estimado en horas y valor monetario que determina el costo de implementar una red IoT de este tipo a nivel vehicular.

CONCLUSIONES

- El desarrollo IoT con tecnología LoRa/LoRAWAN, representa una red vehicular de baja potencia que cubre un alto rango de cobertura. Ésta busca cumplir el objetivo de brindar datos reales al usuario mediante una nueva tecnología implementada en las ciudades de mayor desarrollo tecnológico. Es una opción para, mediante el protocolo LoRaWAN, abrir un mundo más allá de las redes inalámbricas convencionales.
- El presente trabajo permite consolidar el concepto de Ciudades inteligentes, además de la idea de implementar nuevas tecnologías inalámbricas de baja potencia con extensas áreas de cobertura.
- LPWAN, al ser una tendencia y necesidad del mercado en el llamado “internet de las cosas”, tiene por sus estándares de comunicación la posibilidad de ofrecer aplicaciones orientadas a cumplir necesidades básicas en una población; esto en cuanto se adapta a brindar conectividad en zonas de poca accesibilidad inalámbrica.
- La trama que desarrolla la tecnología se encuentra dentro de la capa OSI, siendo una capa orientada a nivel Físico que cuenta con una cabecera MAC y un mensaje con un código de cuatro bytes
- A través de la trama, el preámbulo permite definir parámetros de forma individual para cada paquete. Además, indica el momento preciso para transmitir, el cual a su vez señala el esquema en relación a distancia y el consumo energético que es esencial en la ejecución de una IoT.

- Al implementar los módulos de LoRA y el nodo Shield GPS como un prototipo, se pudo determinar la cobertura que puede ofrecer dicha red e importantes ventajas que brinda para el sector vehicular, optimizando tiempos de recepción en la pérdida de datos que se presentan en tramos de difícil acceso. En este caso, el uso del GPRS, siendo una extensión GSM, permite el acceso múltiple FDMA/TDMA.
- Los datos proporcionados mediante las plataformas virtuales brindan la opción de tener en cualquier momento los datos a través del internet de las cosas y una red segura orientada al usuario.
- ThingSpeak, al ser una plataforma orientada a la IoT, tiene ciertas limitaciones al momento de supervisar varios parámetros, ya que la capacidad de tener varios campos es algo limitante. Mediante el canal creado, se puede originar hasta 6 campos a supervisar, y en este caso la plataforma registra bien los datos pero con ciertos limitantes. La plataforma solo puede supervisar 4 campos dentro de cada canal.
- La página WEB orientada a varios lenguajes de programación, brinda la posibilidad de tener mayores prestaciones en el ámbito de la supervisión de los parámetros de forma dinámica. La opción de tener un resumen de todas las variables almacenadas en el servidor dentro de un solo gráfico, sirve para constatar en un solo campo todos los sensores actuados.
- La ventaja de ThingSpeak radica en tener ambientes públicos y privados. Para gestionar la información adquirida, habilita los procesos de nivel de canal utilizando una llave API, y de esa forma se puede controlar el acceso a datos importantes para que sean privados.
- Al tener un acceso gratuito a un ambiente virtual, se tiene el limitante de que cada dato adquirido se actualice cada 15 segundos; pero en este caso, de tener un usuario pagado, existen errores frecuentes y no se pueden recopilar los datos de la mejor manera.

- La página WEB orientada a varios lenguajes de programación, brinda la posibilidad de tener mayores prestaciones en el ámbito de supervisar los parámetros de forma dinámica. La elección de contar con un compendio del total de las variables recopiladas en el servidor en un solo gráfico, sirve para comparar en un solo ámbito todos los sensores actuados.
- En lo que respecta al diseño y programación de la página WEB, se emplearon una gran cantidad procedimientos de prueba, ya que se manejan 4 variedades de lenguaje, y uno encaminado al servidor SQL. Al mismo tiempo, cada proceso se efectuó tomando como base a un generador de código llamado SUBLIME TEXT.
- El ámbito de la programación tuvo que ser desarrollado en base ARDUINO y equipos Shield de LoRA, esto con el objetivo de alcanzar una comunicación propia con el manejo de los procesos del fabricante. De la misma manera, se creó posicionamiento y lecturas puntuales que causaron inconvenientes al momento de registrar la latitud y longitud del módulo GPS, puesto que *thingSpeak* solo admite 4 campos de lectura.
- Mediante el diagrama del vehículo elegido, se facilitó la programación y coordinación con los sensores montados y acondicionados para la lectura a nivel mecánico, gracias a un registro de parámetros a nivel real automotriz.
- El prototipo implementado, al tener una conectividad sin pérdidas, otorga al usuario la posibilidad de tener un registro a cada momento y en cualquier parte donde se encuentre, ya que los datos no se perderán gracias a una extensión GSM que permite el acceso múltiple a través de la tecnología GPRS.
- Dentro de la configuración de la banda de 868Mhz, se tienen 8 canales definidos.
- Para los parámetros configurables, la tasa de codificación (CR) cuenta con 4 bits útiles en función al valor que se desea codificar, lo cual facilita la

manera de recibir los paquetes para obtener una mejor sensibilidad.

- En el caso del tratamiento de datos a nivel de modos de transmisión, el factor de dispersión (SF) trabaja entre 6 y 12, un parámetro que resulta relevante según el espectro empleado: si fuera mayor, mejora la capacidad de no tener señales parasitas (ruido) en la red.
- A través de pruebas realizadas en espacio abiertos o en línea de vista sin ningún obstáculo que interrumpa la comunicación, tenemos una recepción de datos más rápido que no se demora más de 2 segundos en procesar la información a la WEB, en este caso es una autopista y cada intervalo se demora de 15 segundos a 40 segundos en subir la información adquirida por los parámetros.

RECOMENDACIONES

- Implementar soluciones innovadoras a nivel de la IoT encaminada a un grupo de vehículos o nodos de gran extensión, para así tener una red a mayor escala.
- Implementar y utilizar a la vez diferentes equipos LoRa con una gran variedad de tipos de diferentes fabricantes. La importancia de esto radica en la existencia de inconvenientes al momento de programar cada uno de los módulos.
- Procurar continuar con el desarrollo LoRA en relación a protocolos distintos. Mediante esto, será posible constituir una arquitectura encaminada a redes inalámbricas de una potencia elevada.

LISTA DE REFERENCIAS

- Abarca, P. (21 de 02 de 2017). *SISTEMAS DE CONTROL AUTOMÁTICO*. Obtenido de ASOCIACIÓN DE LA INDUSTRIA ELÉCTRICA-ELECTRÓNICA, AIE: <http://www.aie.cl/files/file/comites/ca/abc/sistemas-de-control-automatico.pdf>
- Alexandra. (30 de 09 de 2017). *Medium* . Obtenido de <https://medium.com/@AlexandraBowen/iot-is-eating-the-world-apis-and-rest-9e0321bc6cbf>
- ALLIANCE, L. (12 de 02 de 2016). *LORAWAN* . Obtenido de <https://www.loraalliance.org/portals/0/documents/whitepapers/LoRaWAN101.pdf>.
- Alliance, L. (14 de 05 de 2017). *EoRAWAN*. Obtenido de <https://lora-alliance.org/lorawan-for-developers>
- ALLIANCE, L. (12 de 08 de 2017). *LORAWAN*. Obtenido de <https://lora-alliance.org/about-lorawan>
- AOGLP. (2016). *¿Qué es GLP?* Obtenido de Asociación Española de Operadores: <http://www.aoglp.com/gpl/que-es/>
- APRENDIENDO ARDUINO. (7 de 03 de 2018). Obtenido de <https://www.aprendiendoarduino.com/category/lora/>
- Archila, D., & Santamaria, A. (14 de Octubre de 2013). Estado del arte de las redes de sensores inalámbricos. . *REVISTA DIGITAL TIA*, 4-14. Obtenido de <https://revistas.udistrital.edu.co/ojs/index.php/tia/article/download/4437/6856>
- arduinoshoppings. (22 de 03 de 2018). Obtenido de <http://www.arduinoshopping.com/producto/shield-lora-gps/>
- Barcos, A. (2015). *Gas Natural Propano GLP Butano Residuales Gas Pobre Manufacturados*. Obtenido de http://www.academia.edu/11115627/Gas_Natural_Propano_GLP_Butano_Residuales_Gas_Pobre_Manufacturados
- Bedoya Martínez, D. A. (2014). *Diseño de un contrastador de temperatura para termómetros de termocuplas tipo S*. Lima.
- Burbano, C. (2017). IMPLEMENTACIÓN DE UNA RED DE SENSORES INALAMBRICOS LPWAN MEDIANTE MODULOS LoRa PARA EL

MONITOREO DE LA CALIDAD DEL AGUA EN 2 RIOS. BOGOTA:
Universidad Distrital Francisco de Cardenas.

- Castellano, I. e. (13 de 02 de 2018). Obtenido de <https://osiberia.org/que-es-una-red-iot-lpwan/>
- Creus Solé. (2009). *Instrumentación Industrial*. Barcelona: Cengage.
- Cristian Toapanta, F. M. (2014). Estudio de una red IP VIDEO EN LA CUIDAD DE MACHALA . QUITO.
- Earnshaw., G. y. (2008). *Química de los elementos (Chemistry of the Elements)*. Obtenido de Lenntech: <http://www.lenntech.es/periodica/elementos/al.htm>
- Ecured. (2005). *Tratamiento Térmico*. Obtenido de https://www.ecured.cu/Tratamiento_t%C3%A9rmico
- Ecured. (2017). *Llama (combustión)*. Obtenido de Química Octavo Grado.: [https://www.ecured.cu/Llama_\(combusti%C3%B3n\)](https://www.ecured.cu/Llama_(combusti%C3%B3n))
- Eduardo Torres, &. B. (2017). *Construcción y modelación de un módulo didáctico de muro Trombe mediante CFD para fomentar los estudios de la calefacción solar pasiva*. Quito: Master's thesis, Escuela Politecnica Nacional.
- Efor. (s.f.). *TECNOLOGÍAS DE COMUNICACIÓN PARA IOT*. Obtenido de <https://www.efor.es/sites/default/files/tecnologias-de-comunicacioon-para-iot.pdf>
- Garces, P. P. (2017). Redes de Areas Extensas.
- García, R. (11 de Noviembre de 2011). *Combustion y Combustibles*. Obtenido de <http://webcache.googleusercontent.com/search?q=cache:xeYm7N6N2NMJ:ki merius.com/app/download/5780666669/Combusti%25C3%25B3n%2By%2B combustibles.pdf+&cd=3&hl=es&ct=clnk&gl=ec>
- Garcia, R. P. (2017). Evauacion de LoRawan para smart city.
- González, M. D. (15 de 05 de 2017). *riunet.upv.es*. Obtenido de <https://riunet.upv.es/bitstream/handle/10251/91597/CARRI%C3%93N%20-%20Desarrollo%20de%20una%20pasarela%20LORA%20y%20evaluaci%C3%B3n%20de%20prestaciones.pdf?sequence=1>
- GUEVARA, R. (12 de 24 de 09). *IoTNET*. Obtenido de <http://www.iotnet.mx/index.php/2017/01/13/sigfox-aprovecha-mejor-el-espectro-radioelectrico-que-lora-por-disk91/>
- Hetpro. (15 de 11 de 2018). *Hetpro*. Obtenido de <https://hetpro-store.com/TUTORIALES/sim900-gsm-shieldarduino/>

Ing.Marques. (14 de 03 de 2017). *newtecnology*. Obtenido de <http://newtecnology.com/online/conexiones-rf>

Instrument, N. (22 de Abril de 2009). *Que es una Red de Sensores inalámbricos*. Obtenido de <http://www.ni.com/white-paper/7142/es/>.

Interempresas. (2017). *Siemens Logo! 8*. Obtenido de <https://www.interempresas.net/Electronica/FeriaVirtual/Producto-Controladores-logicos-inteligentes-Siemens-LOGO-8-129784.html>

INTERNETYA. (26 de 07 de 2018). Obtenido de <https://www.internetya.co/que-es-y-para-que-sirve-una-api/>

IoTenCastellano. (13 de febrero de 2018). *Osiberia*. Obtenido de <https://osiberia.org/que-es-una-red-iot-lpwan/>

LoRa. (7 de 03 de 2018). *Aprendiendoarduino*. Obtenido de <https://www.aprendiendoarduino.com/category/lora/>

LORAWAN. (2018). *THE THINGS NETWORK* . Obtenido de <https://www.thethingsnetwork.org/docs/lorawan/>

Max Weishaupt, G. (2012). Instalacion y operacion del quemador a gas WG20.

Medium. (15 de 11 de 2018). *MEDIUM*. Obtenido de <https://medium.com/beelan/haciendo-iot-con-lora-cap%C3%ADtulo-1-qu%C3%A9-es-lora-y-lorawan-8c08d44208e8>

Mesa, A. (10 de 09 de 2018). *Calculadora de zona de Fresnel* . Obtenido de <http://alvaromesa.com/PIRE/znFresnel.html>

Milton Jami. (2014). *Diseños y construcción de un horno crisol para fundición de aluminio*. Quito: Universidad Politécnica Salesiana.

MULTITECH. (20 de 01 de 2018). *SEMAK* . Obtenido de <https://www.semak.com.ar/tienda/lorawan-que-es>

Nabertherm GmbH. (2016). *Fundición*. Obtenido de Conceptos alternativos de hornos de fundición: http://www.nabertherm.es/produkte/details/es/giesserei_schmelz-und-warmhalteofen

Norbert Hennrich, U. o. (2011). *Industrial burner*. Wisconsin.

Ordoñez, I. (2017). Estudio de la Arquitectura y el nivel de desarrollo de la red LoRa.

Pablo, J. (03 de 12 de 2017). *WAN-LORA*. Obtenido de <http://wan-lora.com/online/conexiones-con-la-tecnologia-loratom/>

- Patagoniatec.* (14 de 03 de 2016). Obtenido de <https://saber.patagoniatec.com/2014/10/gprsgsm-shield-sim900-arduino-argentina-ptec-elecbreaks/>
- Prieto, P. (2008). *Principios Básicos de los PLC*. Obtenido de Observatorio Tecnológico: <http://recursostic.educacion.es/observatorio/web/gl/component/content/article/502-monografico-lenguajes-de-programacion?start=2>
- Rodas, V. (2015). *Diseño e Implementación de un Sistema de Control Automático para seis hornos*. Quito: Escuela Politécnica Nacional.
- Rouse, M. (21 de 09 de 2017). *Search Data Center* . Obtenido de <https://searchdatacenter.techtarget.com/es/definicion/Internet-de-las-cosas-IoT>
- Rouse, M. (21 de 09 de 2017). *Search Data center*. Obtenido de <https://searchdatacenter.techtarget.com/es/definicion/Internet-de-las-cosas-IoT>
- SAP. (12 de 05 de 2016). Obtenido de <https://www.sap.com/latinamerica/trends/internet-of-things.html>
- Semak. (15 de 11 de 2018). *LoRaWAN - ¿Qué es?* Obtenido de <https://www.semak.com.ar/tienda/lorawan-que-es>
- SEMTECH . (13 de 05 de 2016). Obtenido de <https://www.semtech.com/technology/lora/what-is-lora>
- Siemens. (2009). Burner Flame Safeguard Control. *Technical Instructions*.
- Siemens AG corporación. (s.f.). *LOGO! TD Textdisplay*. Obtenido de www.automation.siemens.com/bilddb/search.aspx?objkey=P_ST70_XX_05603
- Sprandley, J. (1980). *Observacion Participante* . Estados Unidos: Wavenland Press Inc.
- Tapia, N. D. (2014). *Transferencia de calor*. Obtenido de <http://www.monografias.com/trabajos15/transf-calor/transf-calor.shtml>
- telecomunicaciones, M. (04 de 10 de 2014). *Mundo teleco*. Obtenido de <http://mundotelecomunicaciones1.blogspot.com/2014/10/zona-de-fresnel.html>
- ThingSpeak*. (19 de 09 de 2018). Obtenido de https://thingspeak.com/channels/577605/private_show

Toapanta, C. (2014). Estudio de una red IoT con video camara en machala . Quito .

Torres, E. (2002). *Construcción de un Módulo Didáctico de Control de Temperatura*. Quito: Escuela Politécnica Nacional.

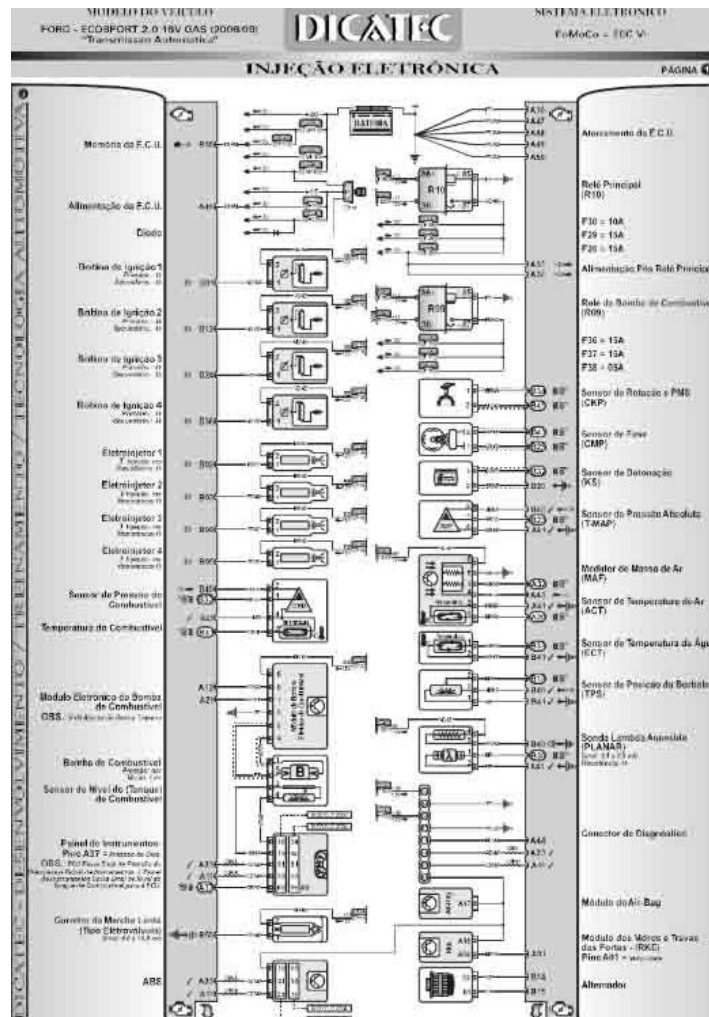
Vega, N. (2013). *Sistema de Control y Monitoreo de Quemadores y Sopladores de la Planta Novacero*. Guayaquil: Universidad Politécnica Salesiana.

wiki.dragino. (26 de 07 de 2016). Obtenido de http://wiki.dragino.com/images/b/b9/Lora_GPS_Shield_Pin_Definition.png

ANEXOS

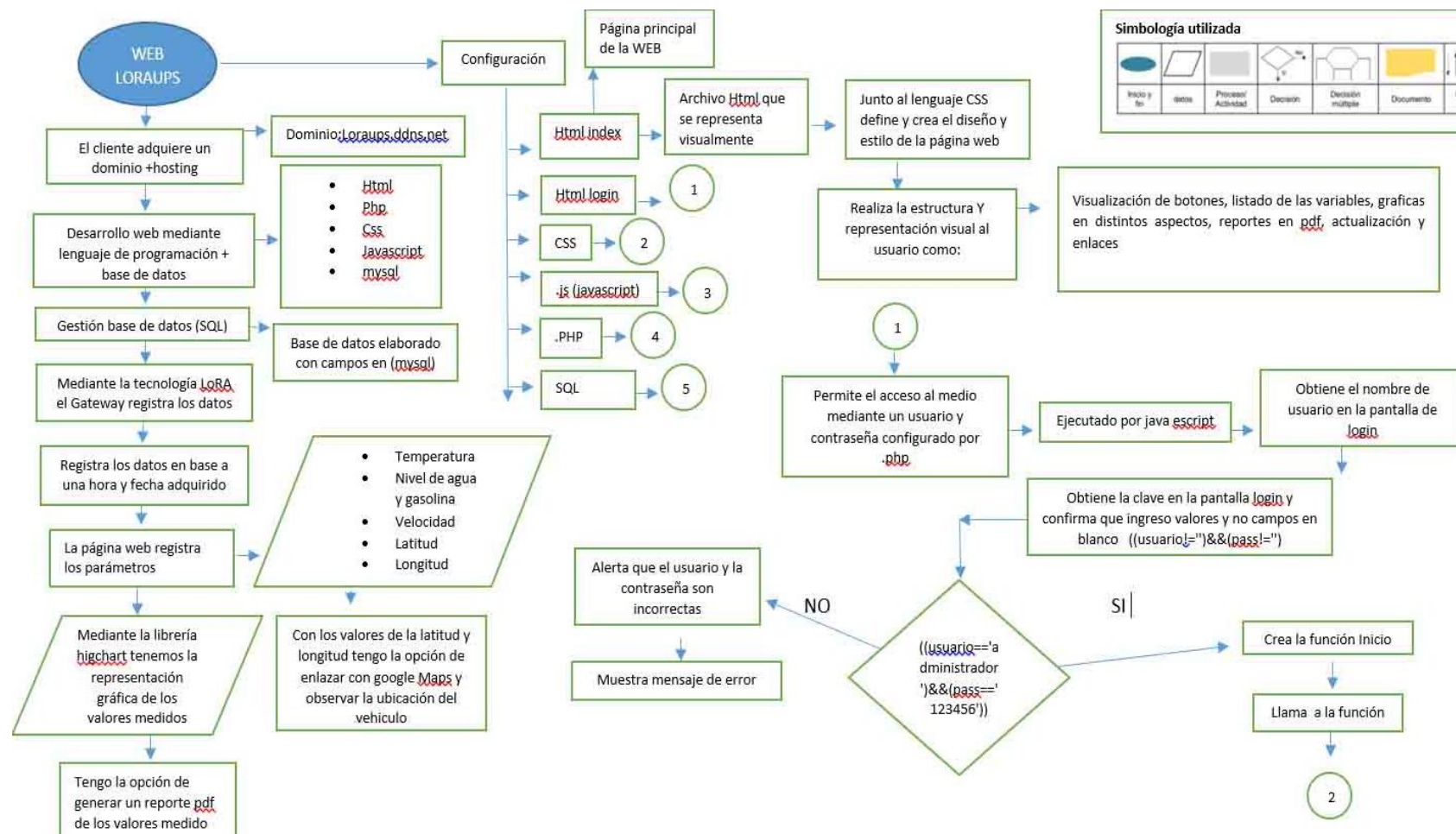
Anexo 1: Diagrama de la computadora del vehículo EcoSport 2006 implementado para la supervisión de los parámetros en la autopista Simón Bolívar

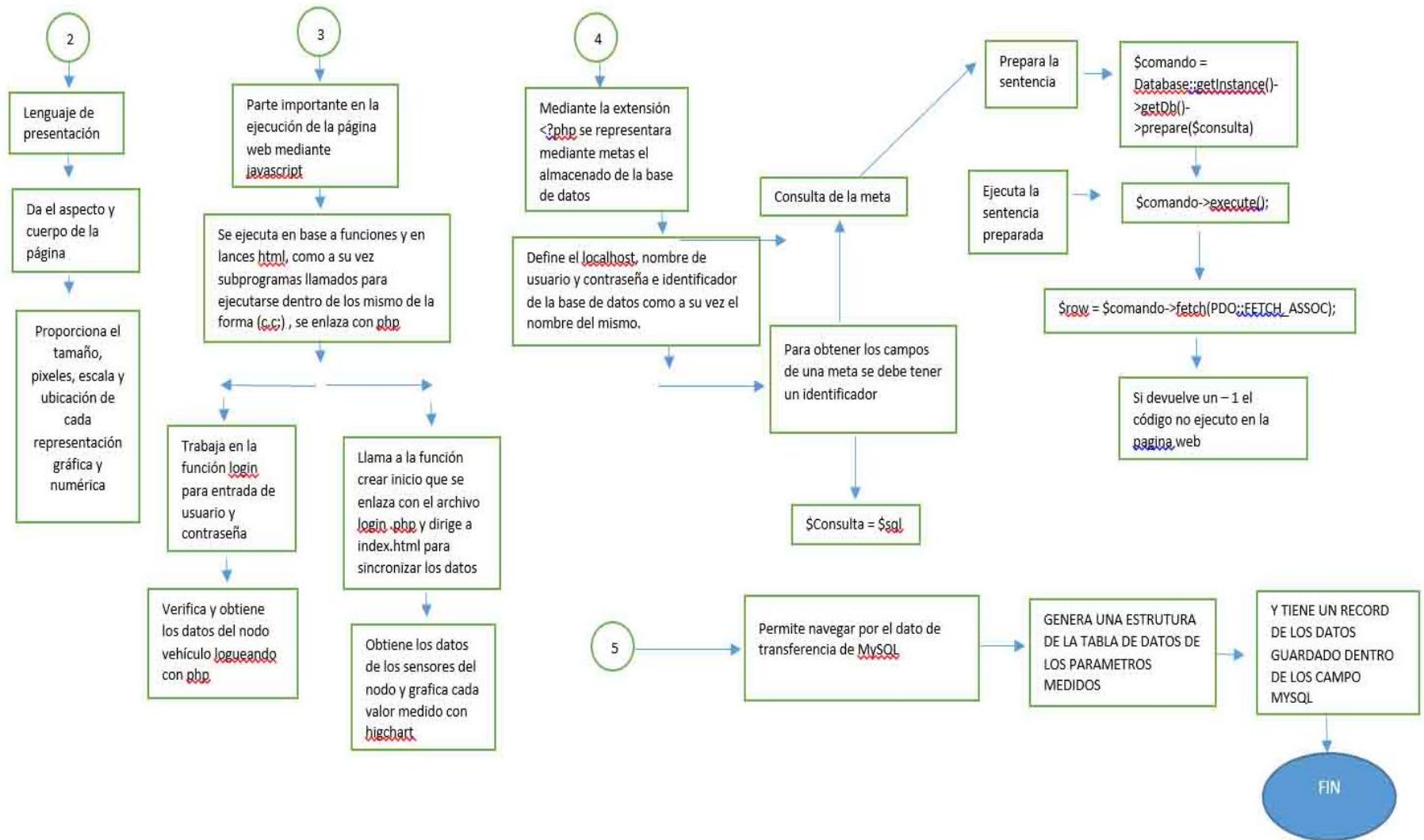
A1.1 Ubicación de los sensores y componentes del EcoSport 2006, diagrama de la ECU (Unidad de control)



A través de este diagrama se procedió a identificar los sensores ubicados en el vehículo y poder realizar las mediciones adecuadas para implementar en la programación y simulación del software como hardware.

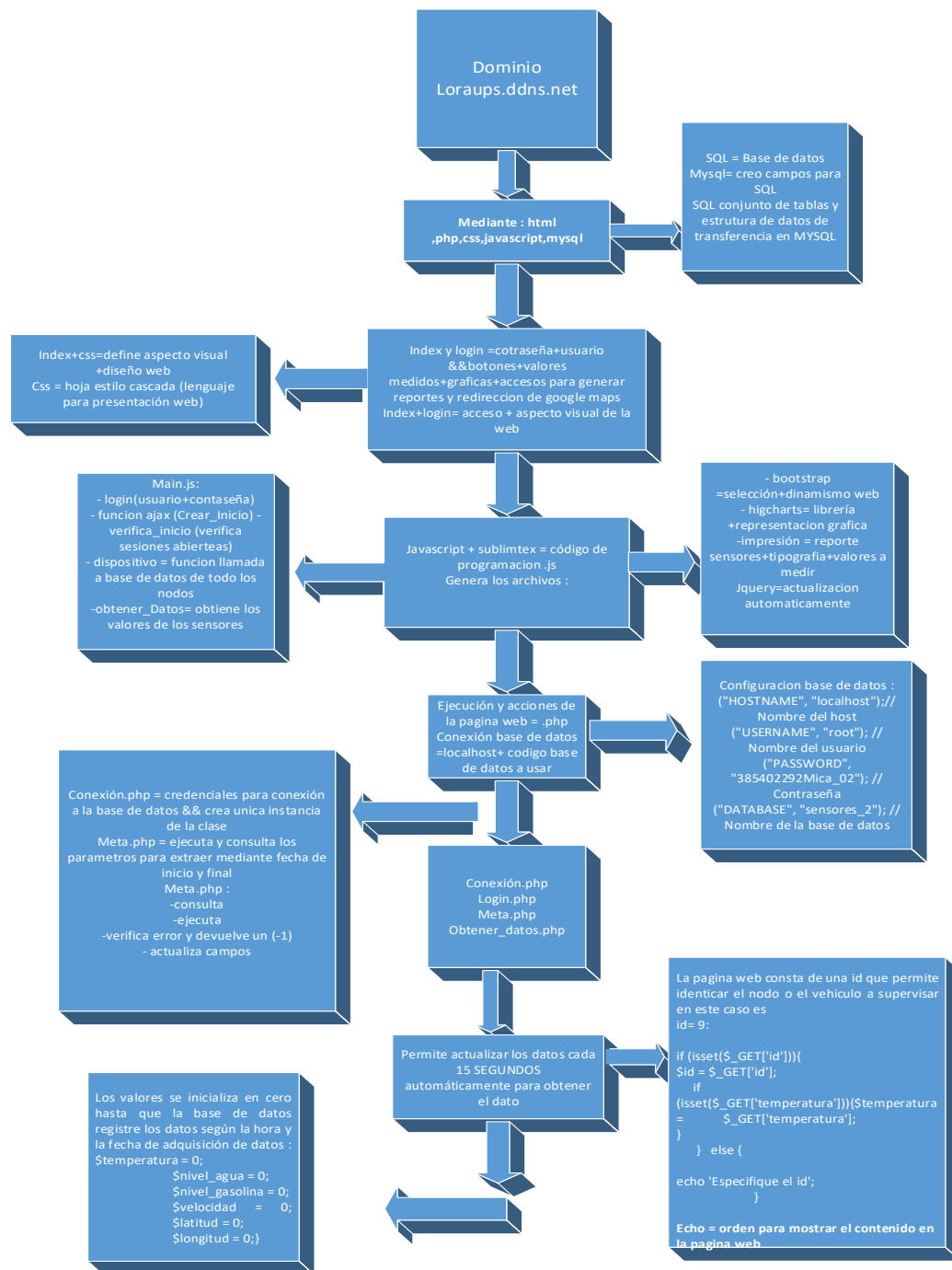
Anexo 2: Diagrama de flujo y diagrama de bloque de la página WEB loraups.ddns.net/index.html





El desarrollo del diagrama de flujo de la página WEB, permitió implementar la página WEB e interactuar con el internet de las cosas (IoT).

A1.2. Diagrama de Bloque del ambiente virtual WEB dnns.net/index.html



A través del diagrama de bloque tengo la posibilidad de ver la acción de los comandos para desarrollar el aspecto y el dinamismo de la WEB.

Anexo 3: Programación Cliente y servidor para la plataforma *ThingSpeak*.

Para realizar la respectiva programación se debe incluir las librerías propias de LoRa y Dragino para poder compilar el programa y tener éxito al subir el programa a la placa ARDUINO UNO y GPS Shield.

Para desarrollar el programa en su totalidad se debe descargar la librería de RadioHead-Master, proporciona una biblioteca para enviar y recibir paquetes mediante radios de datos, esta librería es usada por los módulos LoRa para la conectividad en cliente y servidor. Todo programa realizado se tuvo una base como son los ejemplos que nos proporciona esta librería y a su vez se desarrolló el ambiente del GPRS también con ejemplos proporcionados por los fabricantes, es así que se fue modificando la programación para que trabaje con la tecnología LoRa.

Para compilar correctamente el programa en ARDUINO debe ser la versión 1.8.5.

A1.3. Librería RadioHead-Master en Arduino 1.8.5.



Se debe incluir esta librería que usa los módulos LoRa para enviar y recibir los mensajes en forma de paquetes.

A continuación, se mostrará el código del Cliente compilado para la placa ARDUINO.

dht11_client Arduino 1.8.5

Archivo Editar Programa Herramientas Ayuda



dht11_client \$

```
/*
  Upload Data to IoT Server ThingSpeak (https://thingspeak.com/):
  Support Devices: LoRa Shield + Arduino
  */

#include <SPI.h>
#include <RH_RF95.h>
#include <String.h>

RH_RF95 rf95;

#define dht_dpin A0 // Use A0 pin as Data pin for DHT11.
byte bGlobalErr;
char dht_dat[5]; // Store Sensor Data
String stringOne;
float frequency = 868.0;

void setup()
{
  InitDHT();
  Serial.begin(9600);
  if (!rf95.init())
    Serial.println("init failed");
  // Setup ISM frequency
  rf95.setFrequency(frequency);
  // Setup Power,dBm
  rf95.setTxPower(13);
  // Setup Spreading Factor (6 ~ 12)
  rf95.setSpreadingFactor(7);

  // Setup Bandwidth, option: 7800,10400,15600,20800,31200,41700,
  //Lower BandWidth for longer distance.
  rf95.setSignalBandwidth(125000);

  // Setup Coding Rate:5(4/5),6(4/6),7(4/7),8(4/8)

```

Compilado

dht11_client Arduino 1.8.5

Archivo Editar Programa Herramientas Ayuda



dht11_client \$

```
// Setup Coding Rate:5(4/5),6(4/6),7(4/7),8(4/8)
rf95.setCodingRate4(5);
Serial.println("Humidity and temperature\n\n");
}

void InitDHT()
{
  pinMode(dht_dpin,OUTPUT);//Set A0 to output
  digitalWrite(dht_dpin,HIGH);//Pull high A0
}

//Get Sensor Data
void ReadDHT()
{
  bGlobalErr=0;
  byte dht_in;
  byte i;

  //pinMode(dht_dpin,OUTPUT);
  digitalWrite(dht_dpin,LOW);//Pull Low A0 and send signal
  delay(30);//Delay > 18ms so DHT11 can get the start signal

  digitalWrite(dht_dpin,HIGH);
  delayMicroseconds(40);//Check the high level time to see if the data is 0 or 1
  pinMode(dht_dpin,INPUT);
  // delayMicroseconds(40);
  dht_in=digitalRead(dht_dpin);//Get A0 Status
  // Serial.println(dht_in,DEC);
  if(dht_in){
    bGlobalErr=1;
    return;
  }
  delayMicroseconds(80);//DHT11 send response, pull low A0 80us
  dht_in=digitalRead(dht_dpin);

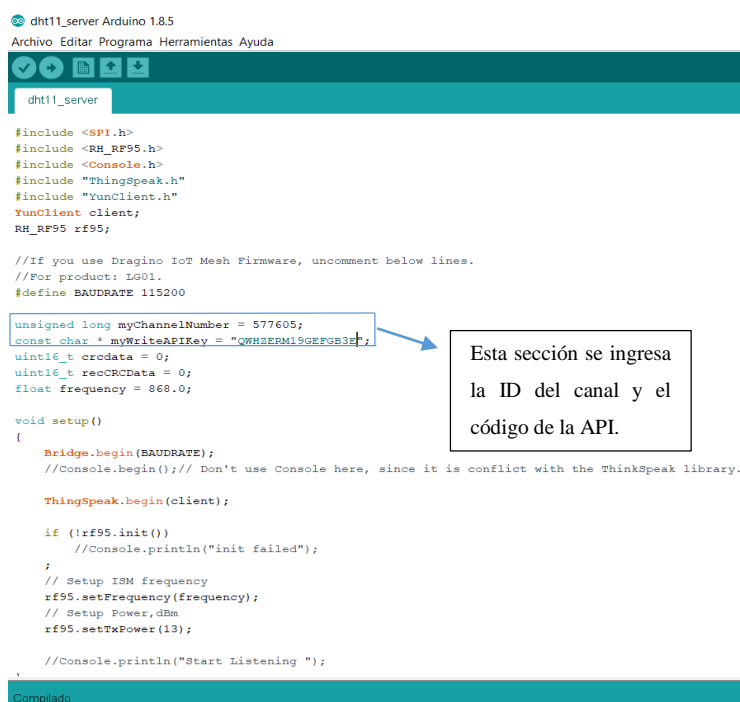
```

Para el programa del servidor se debe ingresar a la plataforma ThingSpeak y tener el código de la ID del canal y la API KEYS, para enlazar con el servidor de la IoT.

En este caso Channel ID: 577605

Key: QWHZERM19GEFGB3E

A1.4. Programa para enlazar con el servidor de la IoT.



```
dht11_server Arduino 1.8.5
Archivo Editar Programa Herramientas Ayuda

dht11_server

#include <SPI.h>
#include <RH_RF95.h>
#include <Console.h>
#include "ThingSpeak.h"
#include "YunClient.h"
YunClient client;
RH_RF95 rf95;

//If you use Dragino IoT Mesh Firmware, uncomment below lines.
//For product: LG01.
#define BAUDRATE 115200

unsigned long myChannelNumber = 577605;
const char * myWriteAPIKey = "QWHZERM19GEFGB3E";
uint16_t crcdata = 0;
uint16_t recCRCData = 0;
float frequency = 868.0;

void setup()
{
  Bridge.begin(BAUDRATE);
  //Console.begin(); // Don't use Console here, since it is conflict with the ThinkSpeak library.

  ThingSpeak.begin(client);

  if (!rf95.init())
    //Console.println("init failed");
  ;
  // Setup ISM frequency
  rf95.setFrequency(frequency);
  // Setup Power, dBm
  rf95.setTxPower(13);

  //Console.println("Start Listening ");
}

Compilado
```

Para la Shield de Lora también se generó un código con el cual se montará al módulo que será como nodo para monitorear el vehículo en movimiento y subir dicho dato a la nube.

A1.5 Código de programación para Shield GPS LoRA.

GPS_client Arduino 1.8.5

Archivo Editar Programa Herramientas Ayuda

```

float frequency = 868.0;
#include <SoftwareSerial.h>
#include <TinyGPS.h>
TinyGPS gps;
SoftwareSerial ss(3, 4); // Arduino RX, TX ,
static void smartdelay(unsigned long ms);
static void print_float(float val, float invalid, int len, int prec);
static void print_int(unsigned long val, unsigned long invalid, int len);
static void print_date(TinyGPS &gps);
static void print_str(const char *str, int len);
void setup()
{
    ss.begin(9600); // SoftSerial port to get GPS data.
    //InitDHT();
    Serial.begin(9600);
    if (!rf95.init())
        Serial.println("init failed");
    // Setup ISM frequency
    rf95.setFrequency(frequency);
    // Setup Power,dBm
    rf95.setTxPower(13);
    // Setup Spreading Factor (6 ~ 12)
    rf95.setSpreadingFactor(7);
    rf95.setSignalBandwidth(125000);
    // Setup Coding Rate:5(4/5),6(4/6),7(4/7),8(4/8)
    rf95.setCodingRate4(5);
    Serial.println("Humidity and temperature\n\n");
    Serial.println("Sats HDOP Latitude Longitude Fix Date Time Date Alt Course Speed Card Distance Course Card Chars Sentences Checksum");
    Serial.println("          (deg)      (deg)      Age      Age (m)    --- from GPS --- ---- to London ---- RX RX      Fail");
    Serial.println("-----");
}
uint16_t calcByte(uint16_t crc, uint8_t b)
{
    uint32_t i;
    crc = crc ^ (uint32_t)b << 8;

```

Este código nos ayuda al que Shield de LoRa en conjunto con arduino procesé y envié datos directamente a la plataforma de ThingSpeak.

A continuación, se desplegará el código que va directamente al Gateway LG01 de Dragino, para su correcto uso dirigirse al manual:

http://www.dragino.com/downloads/downloads/UserManual/LG01_LoRa_Gateway_User_Manual.pdf

Para la conexión directa se debe agregar una librería propia de thingSpeak conocida como (thingspeak-arduino-master).

A1.6 Programación para Gateway LG01.

```

SUBIRATHINGSPEAK_RXSHIELD Arduino 1.8.5
Archivo Editar Programa Herramientas Ayuda

SUBIRATHINGSPEAK_RXSHIELD $

#include <SPI.h>
#include <RH_RF95.h>
#include <Console.h>
#include "ThingSpeak.h"
#include "YunClient.h"
YunClient client;
RH_RF95 rf95;
//If you use Dragino IoT Mesh Firmware, uncomment below lines.
//For product: LG01.
#define BAUDRATE 115200
int led = A2;
unsigned long myChannelNumber = 577605;
const char * myWriteAPIKey = "QWH2ERM19GEFGB3E";
uint16_t crcdata = 0;
uint16_t recCRCData = 0;
float frequency = 868.0;
String stringOne;
String latstr;
String lonstr;
String velostr;
String tempstr;
String aguastr;
String gasostr;
void setup()
{
    Bridge.begin(BAUDRATE);
    //Console.begin(); // Don't use Console here, since it is conflict with the ThinkSpeak library.

    ThingSpeak.begin(client);

    if (!rf95.init())
        //Console.println("init failed");
    ;
    // Setup ISM frequency
    rf95.setFrequency(frequency);

    }

    digitalWrite(led, LOW);

    }
    else
    {
        //Console.println("recv failed");
    }
}
}

void procesar() {
    Console.println("PROCESAR");
    latstr=stringOne.substring(1,7);
    Console.println(latstr);
    lonstr=stringOne.substring(8,15);
    Console.println(lonstr);
    velostr=stringOne.substring(16,18);
    Console.println(velostr);
    tempstr=stringOne.substring(19,21);
    Console.println(tempstr);
    aguastr=stringOne.substring(22,24);
    Console.println(aguastr);
    gasostr=stringOne.substring(25,27);
    Console.println(gasostr);
}

Compilado
```

Para finalizar en el programa podemos evidenciar que se crea una función procesar que lleva la posición de cada parámetro ubicado en la página de THingSpeak

Y de esa forma se ubicará cada parámetro a medir con los sensores.